University of Alberta

On Connectionism

by

István Stephen Norman Berkeley ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Department of Philosophy

Edmonton, Alberta

Spring 1997

National Library of Canada

Acquisitions and Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada

Acquisitions et services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced with the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-21549-0

Canadä

University of Alberta

Library Release Form

Name of Author: István Stephen Norman Berkeley

Title of Thesis: On Connectionism

Degree: Doctor of Philosophy

Year this Degree Granted: 1997

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly, or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

I. Berkeley.

István Stephen Norman Berkeley
4A Bridle Path
Woodcote
Nr. Reading
Berkshire
RG8 OSE
Great Britain

Submitted: 30 January 1997

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled "On Connectionism" submitted by István Stephen Norman Berkeley in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Dr. F. J. Pelletier
(Co-Supervisor)

Dr. M. R. Dawson
(Co-Supervisor)

Wes Cooper
Dr. W. E. Cooper

Mohan Matthen
Dr. M. Matthen

Dr. G. Prideaux

Robert F. Hadley
Dr. R. F. Hadley
(External Examiner)

Date: 23 January 1997

Abstract

This dissertation opens with a discussion and clarification of the Classical Computational Theory of Mind (CCTM). An alleged challenge to this theory, which derives from research into connectionist systems, is then briefly outlined and connectionist systems are introduced in some detail. Several claims which have been made on behalf of connectionist systems are then examined, these purport to support the conclusion that connectionist systems provide better models of the mind and cognitive functioning than the CCTM. It is argued that most claims made on behalf of connectionism often amount to little more than myths. A significant difficulty with standard connectionist research methodology is then described. This difficulty derives from the fact that connectionist systems are seldom subject to detailed analysis and interpretation after training and has serious consequences for the plausibility of connectionist research in cognitive science and as the basis for a challenge to the CCTM. A technique for network analysis is described in detail and the results from the analysis of a particular network are then introduced, in order to show how these difficulties can be overcome. The analyzed network exhibited a number of surprising and intriguing properties. These properties provide the basis for a detailed assessment of the network with respect to the CCTM. There then follows a discussion of the results of the network analysis, with respect to two features which are commonly associated with the CCTM, namely systematicity and compositionality. It is argued that the network has some properties similar to those associated with the CCTM and exhibits, in a weak sense, compositionality and systematicity. It is also argued that the network amounts to a genuinely cognitive theory.

This suggests that there is insufficient evidence at the current time to determine whether or not connectionism presents a genuine challenge to the CCTM. The plausibility of the claim that networks provide the basis of an alternative to the CCTM is then traced, in part, to a revisionist tendency in some contemporary histories of connectionist research. Finally, future research directions in philosophy and cognitive science are suggested.

Acknowledgments

I would like to thank the many people who assisted me with this dissertation. First and foremost, I would like to thank my supervisors. Jeff Pelletier time and time again went well above and beyond the call of duty on all fronts, in addition to providing insightful guidance and encouragement. Mike Dawson (in conjunction with the various members of his Biological Computational Project lab) taught me how to be a cognitive scientist and provided much of the inspiration for the empirical work (and insights into it) described in this dissertation, as well as providing the necessary facilities. I would also like to thank my examination committee, Wes Cooper, Robert Hadley, Bruce Hunter, Mohan Matthen and Gary Prideaux for their many helpful comments. In addition, I must thank the many people who gave me their time and insight whilst I was writing this dissertation. There are too many to list, but Ken Aizawa, Bill Bechtel, Andy Brook, Mason Cash, Andy Clark, Jerry Feldman, Jeff Foss, Steve Giambrone, Charles Ling, Tony Maida, Dave Medler, Ruth Michaels, Marvin Minsky, Zenon Pylyshyn, Don Ross, David Rumelhart, Paul Smolensky and Chris Thornton all deserve special mention. All the errors which remain are all my own work.

Table of Contents

I. Introduction 1

II. The Classical Computational Theory of Mind (CCTM) 7

III. What Is Connectionism? 26
- Introduction 26
- Philosophical Objections to The Classical Computational Theory of Mind 26
- What Is Connectionism? 32
  - Processing Units 33
  - Modifiable Connections 38
  - Learning Rules 40
- Conclusion 42

IV. The Myths of Connectionism 45
- Introduction 45
- Myth 1: *Connectionist systems are biologically plausible* 46
  - Processing Units 47
  - Connections 49
- Myth 2: *Connectionist systems are consistent with real time constraints upon processing* 54
- Myth 3: *Connectionist Systems exhibit graceful degradation* 59
- Myth 4: *Connectionist systems are good generalizers* 63
- Myth 5: *Recent connectionist systems have shown that Minsky and Papert were wrong* 67
- Conclusion 70

V. An Empirical Study: The Interpretation of The Logic Network, L10 72
- Introduction 72
- Rules, Representations and Connectionist Systems 72
- McCloskey's Critique of the Connectionist Research Program 74
- Understanding Trained Connectionist Networks 80
- Bechtel and Abrahamsen's Logic Problem 82
- The Network L10 84
  - The Problem Encoding Scheme of L10 86
  - The Training of L10 88
- The Network Analysis Technique 89
- The Analysis of Network L10 93
  - Definite Features and Inference 'Rules' 100
- Conclusion 104

VI. Connectionist Networks and The CCTM 105
  Introduction 105
  The CCTM Again 105
  Systematicity 107
  Compositionality 118
  Rules and Cognitive Systems 127
  Conclusion 133

VII. Conclusion: Connectionism, Present, Past and Future 136
  Introduction: The Present 136
  History: The Past 140
  Further Research Directions: The Future 150
    Cognitive Science 151
    Philosophy 154

Bibliography 159

List of Tables

Table 2-1: Example of a Turing Machine Table 12

Table 5-1: Examples of Valid Inferences from Bechtel and Abrahamsen's (1991) Logic Problem Set. 83

Table 5-2: The Unary Definite Features Found in Band B of Hidden Unit 4 of Network L10 95

Table 5-3: The Binary Definite Features Found in Band B of Hidden Unit 4 of Network L10 95

Table 5-4: Interpretations of the Bands of the Hidden Units of Network L10 97-98

Table 5-5: The Patterns of Bands Produced by L10 for each Problem Type and the Properties Associated with each Band, as compared to the Properties Associated with the Inference Rules of Natural Deduction 101

Table 6-1: Network Rule for Modus Ponens, as Compared to the Traditional Rule 113

Table 6-2: Network Rule for Type (i) Alternative Syllogism, as Compared to the Traditional Rule 115

List of Figures

Figure 2-1: Diagram of a Turing Machine 11
Figure 2-2: Example of a Maze 22

Figure 3-1: Detail of a Connectionist Processing Unit 33
Figure 3-2: A Step Activation Function 34
Figure 3-3: A Sigmoidal Activation Function 36
Figure 3-4: A Gaussian Activation Function 37
Figure 3-5: Detail of a Connection Weight 38
Figure 3-6: Layers of Connections and Weights in a Connectionist Network 39

Figure 5-1: The Structure and Encoding Scheme of the Network L10 85
Figure 5-2: An Example of a Jittered Density Plot 90
Figure 5-3: An Example Jittered Density Plot of a Hidden Value Unit 91
Figure 5-4: Jittered Density Plots for the 10 Hidden Units of Network L10 93
Figure 5-5: The Jittered Density Plot of Hidden Unit 4 of Network L10 94

I

Introduction

Issues arising from the computational view of the mind will be central to this dissertation. There is a long tradition of philosophers taking inspiration for their philosophical theorizing from mechanical devices. This was as true at the time of Descartes, as it is today (see Meyering 1989). One particular contemporary instance of this tendency is the consideration of computational systems as a structuring metaphor for the mind (see Boden 1981). This structuring metaphor has inspired an ambitious research program in cognitive science which, in turn, has inspired a considerable body of philosophical theorizing about the mind. Central to both avenues of research are questions concerning representations or tokens. These questions arise as a direct consequence of the adoption of the computational metaphor. There are certain presumptions that have been made about the nature of representation or tokens, which have been central to both the philosophical and technical research programs. These presumptions define what has been called (e.g. by Fodor and Pylyshyn 1988) 'classical' cognitive science and philosophical theorizing about the mind. However, recently these suppositions have come under attack. Advocates of the 'connectionist' approach have claimed that these suppositions are incorrect. Although it is usually assumed that there is some kind of incompatibility between the connectionist and classical positions, I will argue in this dissertation that these incompatibilities are not as great as has been assumed.

In the chapter which immediately follows this one, introduce a paradigm example of a computational device, the Turing machine. Turing machines have played a significant role in philosophical theorizing about the mind, especially with respect to functionalism. This Turing machine-like conception of the mind has (in conjunction with other factors, such as Fodor's (1975) language of thought hypothesis) underwritten a view of the mind, which I term the 'classical computational theory of mind' (CCTM), which is central to much of both contemporary philosophy of mind and cognitive science. In this chapter I attempt to consider in detail the properties which are supposed by the CCTM to be shared by both Turing machines and minds. The purpose here is to develop a clear picture of the position, so that the alleged challenge to the CCTM posed by connectionist systems can be assessed.

Although the CCTM is well established in the philosophical literature, it has not been without its critics. Philosophers, such as Searle (1980), and Dreyfus (1991), have argued on a number of grounds that this position is deeply flawed. A further challenge to the CCTM has come about due to the increase in interest in what are known as 'connectionist' style models. Both Dreyfus and Searle are cautiously optimistic that this style of model may offer a means of answering their objections to the CCTM. The third chapter will briefly introduce and discuss the positions of Dreyfus and Searle, with respect to the CCTM and the alleged connectionist alternative to it. However, the bulk of the third chapter will be taken up by a detailed description of the major features and components of connectionist models, concentrating especially upon the class of such models which undergo training. In the conclusion of this chapter, a number of strong and philosophically significant claims that have been made about connectionist systems, especially with respect to the CCTM, will be introduced.

The purpose of chapters two and three is to set out the classical and connectionist positions which, it has widely been supposed, are in opposition to one another. This provides a basis upon which a more careful assessment of the alleged conflict can take place. One of the problems with the literature on connectionism is that, although there is quite widespread agreement that connectionist systems offer *some* kind of challenge to the traditional position, the details and precise nature of the challenge is not always explicit. This is, in large part, due to the fact that connectionist theorists usually define their positions negatively, by saying in what respects they disagree with more traditional positions.

The fourth chapter involves a discussion of a number of claims which have appeared in the connectionist literature that provide (in part) an explanation of why philosophers of mind have become so interested in connectionist systems. Most of the claims concern, either directly or indirectly, the relationship between connectionist systems and the more traditional, classical ones. There are a number of respects in which the empirical adequacy of the traditional systems has been called into question. It has been claimed by connectionists that their models have properties which enable them to address these issues. If this claim were true, then there would be good grounds for believing that such models provided the basis for a better account of human cognitive functioning and, as a consequence, would also provide a sounder basis for philosophical theorizing about the mind. Unfortunately, as I will argue in this chapter, none of these claims are as unproblematic as their advocates would have us believe. Even where there are some virtues to the claims in certain instances, the claims require careful qualification that is seldom given.

Chapter five opens with a brief discussion of a further myth about connectionist systems, concerning the representational structures and the operations which manipulate those structures. This myth is of particular significance as, were it to be true, it would provide strong grounds for maintaining that there were significant differences between the CCTM and a theory of mind based upon connectionist systems. However, as with the myths discussed in the previous chapter, there are significant problems with this claim. In fact, consideration of this myth leads to an even more fundamental difficulty with connectionist systems, if they are supposed to inform cognitive theorizing. This difficulty arises, in part, because of the complexity connectionist systems after training and, in part, because of problems with standard connectionist methodology. The main body of this chapter though will be taken up with the detailed analysis and interpretation of a particular connectionist network which was trained upon a logic problem originally studied by Bechtel and Abrahamsen (1991). The purpose of undertaking this analysis is to show that, in at least one instance, when connectionist systems are subject to detailed scrutiny, they reveal properties (contrary to standard expectations) which, superficially at least, appear to be surprisingly similar to those usually associated with systems which are consistent with the CCTM.

Chapter six is focussed upon trying to draw firm philosophical conclusions, based upon the proceeding chapters. In particular, I attempt to assess the extent to which the network introduced in chapter five has the properties associated with the CCTM (which were introduced in chapter two). It turns out, contrary to what might be initially expected, that the network has a significant number of the properties associated with the CCTM. This is not to say though that networks are entirely consistent with the CCTM in all respects though. These differences become especially apparent when considering the extent to which the network can be said to exhibit the properties of systematicity and compositionality. The extent to which the network can have these properties ascribed to it is somewhat limited, due to the nature of the task upon which it was trained. However, the detailed analysis of the network does serve to show that the network is cognitive in nature, contrary to what might be expected, given the claims of Fodor and Pylyshyn (1988). The results also suggest that many of the claims which have made about networks, with respect to the CCTM are much too simplistic, as the relationship between networks and the CCTM is far more subtle and complex than is generally assumed. This being the case, the straightforward claim that networks can form the basis of an alternative to the CCTM is rejected.

In the concluding chapter, chapter seven, the results from the discussion of the previous chapters are reviewed. A brief discussion of the history of network research is then offered and it is argued that many modern histories of network research have over emphasized the antagonism between the network and more traditional research programs, at the expense of historical accuracy. This helps to explain, in part, why it is that the differences between connectionist systems and the CCTM have become exaggerated. This chapter concludes with a few suggestions about future directions of research. In particular, a number of projects which fall within the scope of cognitive science are suggested, as well as some complementary avenues of philosophical research.

II

The Classical Computational Theory of Mind (CCTM)

There is a long tradition of philosophers taking inspiration from mechanical and devices and technological innovation. Consider for example, Descartes' analogy of two clocks, offered in the *Principles* (IV, CCIV) in order to illustrate how superficially similar things may have different causes, or the cosmology of *La Monde* being based upon optical theory (see Meyering 1989). This tendency continues today. In contemporary philosophical literature, it is common to find computational systems being used as a structuring metaphor for the mind, especially in so-called functionalist theories (see Boden 1981 for a discussion of the origins of this metaphor).[1] This view of the mind has close affiliations and links with such philosophical positions as "The Language of Thought Hypothesis" and has encouraged an ambitious research program in cognitive science, which in turn has inspired a considerable body of philosophical theorizing. Although these further ramifications of the metaphor are not the focus of this dissertation, it is useful to note that a whole tradition has been generated from the starting point under consideration here.

One problem is that the exact relationship between computers and minds is not entirely clear. The difficulty with the computer metaphor[2] is knowing exactly which property or properties of the base domain (in this case, computers) is shared by the target domain (in this case, minds). The metaphor has, despite its shortcomings, nonetheless provided the basis of a significant position in the philosophy of mind and cognitive science, known as the 'Computational Theory of Mind' (CTM) or 'The Computational Theory of Cognition' (CTC).

---

[1] Some (e.g. Pylyshyn 1984) believe more strongly that this not a metaphor at all. For these theorists, slogans such as 'cognition is computation' are taken as literal statements.

[2] Or 'computer analogy', if one wishes to make a technical distinction between analogies and metaphors. Since such a technical distinction will not play a role in the discussion that follows, the two terms will be used interchangably here.

Very roughly, the CTM is based upon the idea that cognition of any sort is just information processing. The scope of this thesis is not always clear, insomuch as there are grounds to be somewhat reticent about the strong claim that *all* mental phenomena are also computational phenomena. Indeed, there also may be grounds for wondering whether all mental phenomena are truly cognitive phenomena. However, provided that either the class of cognitive phenomena is such that it includes a sizable portion of mental phenomena, or that most mental phenomena arise from a common set of mechanisms, then this particular issue need not be a concern here. However, even if it turns out that only a portion of mental or cognitive phenomena are computational, this thesis is still substantial, so long as it provides a means of studying a (significant) sub-set of mental states.

The assumption that cognition is (in large part) computational in nature is foundational to the discipline of cognitive science. It is also important in the philosophy of mind (see for example Cummins 1989, Sterelny 1990 and Lloyd 1989). Cummins (1989: p. 13) gives a fairly typical philosophical formulation of the thesis, when he remarks that,

> By computational theories of cognition I mean *orthodox* computational theories--theories that assume that cognitive systems are automatic interpreted formal systems...i.e. that cognition is disciplined symbol manipulation.[3]

---

[3] Putatively 'non-orthodox' theories will be the topic of the next chapter.

As a matter of fact though, it is quite difficult to give a definitive detailed account of the computational theory of mind. This is because the position is so well known that it has been formulated many times in many different contexts, and there is no one single veridical version.[4] Moreover, it is neither entirely clear exactly which properties of computational devices are relevant to the CTM, nor are the terms in which the CTM is stated uniform and unambiguous with respect to how exactly they relate to computational devices. Another difficulty with the CTM is that it is far from clear exactly which type of computational device is the basis for the analogy. Clarification with respect to these issues is necessary, if a clear picture of the properties essential to the CTM is to be developed. This is important not only with respect to getting the position correct, but also for accurately assessing the degree to which (and in which respects), allegedly 'non-orthodox' theories differ from the standard position.

One way to explore in detail the metaphor which underlies the CTM, and thereby clarify the position, is to consider a paradigm example of a computational device, in the philosophical literature. The metaphor between minds and computers can be traced back to a paper by Turing (1950).[5] Since Turing's (1950) argument is (historically, at least) the original source of the metaphor that underlies the CTM, it is appropriate to begin clarifying the CTM by looking at the properties which Turing suggested as being essential to computers and then looking more specifically at a kind of device which exhibits the relevant properties. This (hopefully) will make the properties of the kinds of machines which form the metaphorical base of the classical (or 'orthodox') CTM explicit, and thereby help to determine more precisely the properties which minds and machines are supposed to have in common. From this point on, I will refer to the Classical version of the CTM as the 'CCTM'.

---

[4] See for example Fodor (1975), Field (1978), Schiffer (1981), Stich (1983), Pylyshyn (1984), Haugeland (1985), Cummins (1989), Sterelny (1990) and Searle (1992) to cite just a few examples.

[5] It is interesting to note that, according to Boden (1981: p. 31), the same relationship did not explicitly appear in the psychological literature until ten years later.

Before proceeding to the exploration of the metaphor, I should make it clear that I will not be addressing many of the traditional issues in the philosophy of mind, such as intentionality, consciousness, qualia and so on. The focus instead will be limited to the portion of mentality which has been classically associated with Turing machines. It might be objected that this focus really amounts to the consideration of methods of problem solving in general (which involve considerations broader than those directly connected to the mind); however, the (alleged) contrast between classical verses connectionist positions still arises even on this construal. Since philosophy of mind is the arena in which the most substantial claims in the literature have arisen, I will continue to cast the contrast in those terms.[6]

Turing (1950: p. 437) describes computational devices as having three parts. These are, in Turing's terminology, a 'store', an 'executive unit' and a 'control'. The 'store' is a mechanism whereby information can be held and retrieved. Turing (1950: p. 437) explicitly thinks of this as being analogous to human memory. The 'executive unit' is the part of the device which actually carries out the various operations which take place upon the information which the computational device operates upon. Finally, the 'control' is the part or property of the device which ensures that the operations performed upon information are performed correctly.

---

[6] I might also remark that the position that I favour with respect to theories of mind is some form of functionalism allied with some sort of syntactic theory of mind, as will become apparent in the discussion of this and following chapters (Cf. Fodor 1975, Stitch 1983, or Sterelny 1990).

One computational device which has all the parts Turing describes is the device which bears his name: a Turing machine. A Turing machine is a simple, yet very powerful computational device (for the original description of Turing machines and their power, see Turing 1937, a more accessible account is provided by Hopcroft and Ullman (1979)).[7] Turing machines consist of two basic components. The first of these is a tape which is divided into discrete regions. Each of these regions can contain just one token from a finite alphabet. In principle, the tape can be unlimited in length. The tape acts as the Turing machine's 'store'. The second major component is known as the 'head'. It acts as the machines 'executive unit'. The head can move backwards and forwards along the tape, one square at a time, and read, write and erase items on the tape.

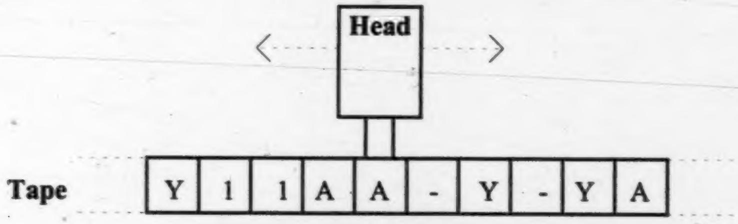

Figure 2-1
Diagram of a Turing Machine

The other important property of the head is that it can adopt internal states. The internal state of the head at any particular time t_1 is dependent upon two factors; (a) the symbol that the head is scanning at t_1, and (b) the previous internal state of the head at time t_0. The set of internal states that the head of a Turing machine can assume is finite and is determined by what is known as the 'machine table' of the head. The machine table also specifies the movement and read/write/erase operations of the head. It acts as the 'executive control' of the Turing machine.

---

[7] For a non-mathematical treatment of Turing machines, see Haugeland 1985: pp. 133-140

Here is an example of a machine table, which comes from Haugeland (1985: p. 139);

		Internal State of The Head						
		1	2	3	4	5	6	7
Symbol on Tape	Y	-L1	-L1	YL3	YL4	YR5	YR6	-R7
	-	-L1	YR2	HALT	YR5	YL3	AL3	YR6
	1	1L2	AR2	AL3	1L7	AR5	AR6	1R7
	A	1L1	YR6	1L4	1L4	1R5	1R6	-R2

Table 2-1
Example of a Turing Machine Table

This machine table describes a Turing machine which can be in seven distinct states and has a vocabulary of just four items (Y, -, 1 and A). Each cell in the machine table specifies precisely the consequent actions of the head in terms of what it should write onto the tape, the direction which it should move along the tape, and the next internal state to adopt.[8]

Consider for example, what would happen if the machine with the above table were in state 3 and the head was over an "A" on the tape. Under these circumstances, the head would execute the command "1L4". That is to say, the head would write a "1" on the tape, move one square to the left and adopt internal state 4. The response of the machine to every possible situation (given the finite vocabulary and number of internal states) is determined by (and predictable from) the machine table. It is also important to note that (for a Turing machine with the machine table illustrated above) if the machine is in internal state 3 and the symbol on the tape beneath the head is a "-", then the instruction to the machine is "HALT". This special instruction is necessary in order to stop the machine when it has completed executing a particular routine.

---

[8] As a matter of fact, a table of this kind is only one particular way of specifying the internal states of the machine. All that is required is that whatever plays the role of the head in the machine be equivalent to some finite state automaton.

It is perhaps worth pausing briefly to wonder why the thesis that 'cognition is computational' might be thought to be even remotely plausible at all. The answer to this question comes in part from certain facts originally shown to be true of Turing machines (and equivalent devices). Turing (1937) proved that there is a special class of Turing machines, which are known as 'Universal Turing Machines' which can simulate any other Turing machine.[9] When this fact is combined with what is known as the Church-Turing thesis, it serves to build a (potential) link between cognition and computation. One formulation of the Church-Turing thesis (due to Minsky 1963: p. 108) states that,

> Any process which could naturally be called an effective procedure can be realized by a Turing machine.

Now, assuming that at least some cognitive processes of biological agents can be described as 'effective procedures', it follows that a universal Turing machine can realize them. Moreover, there *do* seem to be some cognitive processes of biological agents which can be described as 'effective procedures'. For example, effective procedures for mathematical operations such as addition or multiplication can be so given. Thus, it is for these reasons that it is supposed that Turing machines can provide a link between computational devices and cognitive abilities.

---

[9] As a matter of fact, the machine table illustrated is a machine table for one of the simplest universal Turing machines. If this machine table is supplied with a tape of infinite length, then the resulting machine would be 'universal' in Turing's (1937) sense. For further details see Haugeland (1985: p. 139).

Strictly speaking, Turing machines are mathematical objects. However, this presents something of a problem for the analogy between Turing machines and minds. If the metaphor of the CCTM is supposed to be between real minds and an abstract, mathematical object, then this is either a very poor one (as abstract items and 'embodied' items have very few properties in common), or it is not really a case of a metaphor at all. In this latter case, rather than being a true metaphor, it would be something akin to a scientific hypothesis, whereby the Turing machine is taken as a 'model' of the mind. In order for the CCTM to be based upon a proper metaphor, the metaphor would have to hold between minds and *actual* (or 'instantiated') Turing machines. This is a point seldom noted in the literature on the computational metaphor (see Boden 1981), although it makes a difference when it comes to the way that the CCTM is understood (especially with respect to property (1), below).

Both Turing machines and minds, as conceived under the CCTM, are supposed to share a number of properties. I will list and discuss each of these properties in turn. Turing machines and minds share the property of having:

(1) A finite set of discrete tokens.[10]

In a Turing machine, such as the one illustrated above, just one token can be contained within each square of the machine tape. If a Turing machine is considered as a mathematical entity, then by definition, these tokens are discrete from one another. It is only in the case of embodied Turing machines that this becomes an issue, as it is necessary that the machine's head actually be able to distinguish tokens from one another. The number of basic tokens (and the number of internal states) must be finite, as there has to be a row of the machine table which specifies what the machine will do when it reads each token, for each of the possible states of the machine head. Of course, this does not preclude complex strings of tokens being constructed from the set of more basic ones. Note also that there are no in principle constraints upon the kinds of things which may act as tokens. The only condition which needs to be satisfied is that the tokens are distinguishable from one another by the mechanism of the Turing machine.

---

[10] It is common for these tokens to be simply called 'symbols'. However, I prefer the term 'tokens' as it is more neutral.

The CCTM as traditionally interpreted, also supposes that Turing machines and minds share:

(2) A capacity to store and retrieve sequences of tokens.

In a Turing machine, the tape provides a means of storing strings of tokens from the Turing machine's finite vocabulary. The tape also provides a means by which tokens can be retrieved after storage. These properties are important to the functioning of Turing machines, as it is the tape alone (or whatever acts as what Turing (1950) calls the 'store') which enables such machines to have the ability to store information over time.

There are reasons to believe that cognitive agents have properties which are at least analogous to those of Turing machines in this respect. It certainly appears that biological cognitive agents have an ability to store information (i.e. have memories). Likewise, information can be stored in the sequences of tokens upon the tape of a Turing machine. Moreover, Turing (1950: p. 437) explicitly equates the store of a computational device with the memory of human agents. However, although the analogy is appealing in some respects, there are also some significant differences between Turing machines and human beings. For example, there is a good deal of empirical evidence which suggests that human memory is divided into short and long term storage components (See Best 1986: pp. 111-220), as well as sub-components which handle different types of information. No such mechanism are to be found in most common computational functional architectures. This suggests that although Turing machines and minds may share the property of being able to store and retrieve tokens (and thereby information), no inferences can be directly drawn about *how* this is achieved in each case, so this is a point at which the analogy between minds and machines is a little strained.

The third property which the CCTM supposes is shared by both Turing machines and cognitive agents is:

> (3) A capacity to perform a determinate set of precise and exceptionless operations upon tokens.

The operations that a particular Turing machine performs are exceptionless, determinate and predictable. This is because they apply in every case that a particular machine is in, or subject to, a particular input token/head state combination.

Although any particular Turing machine's table of operations is necessarily finite, these operations may be utilized to compute complex algorithms by the appropriate strings of symbols being placed upon the machine's tape. Thus, from a finite set of resources, in terms of head states, tokens and operations, considerable power can be gained, with respect to the algorithms which can be computed. Indeed, it is the capacity of Turing machines to do just this, which enables them to compute any particular computable algorithm, when supplied with the appropriate machine table, amount of tape and initial starting conditions (i.e. tokens upon the tape).

These three properties have a number of consequences which also have been influential upon the CCTM. For example, given that the tokens of (1) are located in the tape, and that the operations of (3) are located in the machine head, it follows that Turing machines support a sharp and principled distinction between the tokens involved in computation and the operations which manipulate those tokens. Thus, a further property of Turing machines is that they have:

> (4) A capacity to support a principled distinction between tokens
> and operations which manipulate those tokens.

Moreover, this property is often believed to be important to the CCTM (see Cummins' 1989 formulation, quoted above).

Another property which is important for the CCTM is that Turing machines have a capacity to place tokens adjacent to one another on the tape, so that they form complex or compound strings, composed out of the individual tokens which have operations defined for them in the machine table. Notice though, that there is a difference between this capacity and the previous ones discussed. Turing machines just operate upon individual tokens.[11] Although the results of successive operations may be complex strings of tokens on the tape, it requires some external observer to notice this complexity. So in some sense, this capacity is something like a Lockean 'secondary property' of a Turing machine. Nonetheless, this property is important for the CCTM. The reason that this is important is that it provides a means by which strings of arbitrary length can arise upon the tape of a Turing machine, even though a Turing machine has only finite resources in terms of the tokens which have operations defined for them in the machine table (cf. Property (3) above). A closely related 'secondary property' of strings of tokens on a Turing machine tape is that they can have *structural* complexity.[12] Turing machines, then, have:

(5) A capacity to construct structurally complex strings of tokens.

Having this property guarantees that Turing machines can be interpreted as computing algorithms which operate upon structurally complex strings of tokens. The reason this is of significance to the CCTM is that human beings appear to have a analogous capacities. Consider the case of language. In a language the individual words of the language are the tokens and strings of words (i.e. phrases and sentences) and have structural complexity, due to the grammar of the language. Human beings can construct structurally complex strings of linguistic tokens (i.e. grammatical phrases and sentences), thus human beings too can be interpreted as sharing this ability with Turing machines.

---

[11] Strictly speaking, this is not true of all possible Turing machines, as Turing machines can have more than one head (see Hopcroft and Ullman 1979: pp. 154-163). However, it *is* true of the prototypical Turing machine, which I am concerned with here.

[12] An interesting twist which arises with respect to the question of complex tokens, as they have been treated in the recent literature, is an appeal to the principle that 'in order to token a complex token, it is necessary to token its atomic parts' (Fodor and Pylyshyn 1988). Strictly speaking, this principle is not required by the basic version of the CCTM, but instead might be thought of as being an additional, eighth, condition of the CCTM.

The fact that Turing machines can be interpreted as producing structurally complex strings of tokens, is closely related to yet another secondary property of Turing machines which is also important to the CCTM. This property is:

> (6) A capacity to differentially perform operations upon structurally complex strings of tokens, dependent upon the order of the tokens in the string.

In some ways, the fact that this property can be ascribed to Turing machines is not especially surprising, given that the precise nature of each individual operation is a function of the token on the tape beneath the machine head and the state of the machine head. However, the fact that the states of the machine's head are determined by the previous operations performed means that the same set of tokens in two strings can, by being ordered differently, produce very different results. It is the order of the tokens in the string which makes the crucial difference.

Property (6) of Turing machines is closely related to one of the most distinctive features of the CCTM. The CCTM (for example Cummins's 1989 version, quoted above) makes an appeal to the notion of 'formality' (or often 'syntax'). The 'formality condition' (Fodor 1980: p. 63) is the requirement that,

> ...mental processes have access only to formal (nonsemantic) properties of the mental representations over which they are defined.

A formal system is a system which consists of a specified set of tokens and rules for operating upon those tokens (see Martin 1991: p. 90). Fodor (at least in 1980, p. 65), seems to endorse this view, as is evidenced by his comment that,

> I take it that computational processes are...formal because they apply to representations in virtue of (roughly) the *syntax* of the representations.

Turing machines are clearly interpretable as being formal in this sense, at least with respect to the tokens for which rules are defined in the machine table. The fact which makes plausible the attribution of property (6), namely that previous operations have an influence upon subsequent operations, also means that Turing machines can be interpreted as satisfying a formality condition for structurally complex strings of tokens.

A further significant, though secondary, property which can be ascribed to Turing machines (and which is commonly incorporated into the CCTM), is that the strings of tokens upon the tape of a Turing machine (under the appropriate circumstances) can be interpreted as 'standing for' various other things, including mental entities of various kinds. Fodor (1980: p. 65) is fairly explicit about the importance of this fact when he notes that,

> ...[W]e will think of the mind as carrying out whatever symbol manipulations are constitutive of the hypothesized computational processes. To a first approximation, we may thus construe mental operations as pretty directly analogous to those of a Turing machine. There is, for example, a working memory (corresponding to a tape) and there are capacities for scanning and altering the contents of memory (corresponding to the operations of reading and writing to the tape)....If mental processes are formal, they have access only to the formal properties of such representations...[13]

Interpretability then is the crucial property of Turing machines, which makes the analogical link between such machines and minds. Furthermore, the property of Turing machines, that they can:

---

[13] Notice Fodor's use of phrases such as "...we will think of the mind..." and terms like "...construe...". This I take to be indicative of the fact that the properties are 'secondary'.

(7) support principled interpretations

is also important for the CCTM, at least as conceived of by Fodor (1980). Although it is debatable whether or not mental representations themselves really are such that they are the kind of items which can be subject to 'principled interpretations', a commitment to the CCTM brings along with it a presumption that mental representations are also interpretable in the relevant manner (see Haugeland 1985: p. 100). This is a situation where the analogy between minds and machines attributes properties of machines to minds.[14]

It is particularly important (at least in the case of Turing machines) that interpretations be 'principled'. This is because without this additional condition, the point becomes trivial. In the case of a Turing machine, an interpretation specifies a set of mapping relations between strings of tokens on the Turing machine tape, and some other set of states (such as mental states). An interpretation is principled just in case it provides a coherent mapping from the strings of tokens to the other set of states.[15] It is worth considering an example of interpreted tokens in a Turing machine, as this will serve to clarify the idea and ward off some possible confusions. However, before doing this, a few words are in order about the source of 'interpretation'.

Although a discussion of interpretation is commonly found in associated with discussions of the CCTM, by an large, these are far from satisfactory. This is because, it is unclear where interpretations come from. For example, Haugeland (1985: pp. 87-123) discusses interpretation at length, without being able to resolve this question. Instead he simply refers to it as "the mystery of original meaning". Similarly, Cummins' (1989) discussion of the topic is, by his own admission, incomplete.[16] Given these facts, I will not even try and develop an answer to the 'mystery of original meaning', here. However, given that the clearest case is one in which some observer attributes an interpretation (i.e. the interpretation is a secondary property) to a Turing machine, this is the kind of case I will consider. I will attempt to remain neutral with respect to other possibilities.

---

[14] There are a number of complex issues which arise here. For a detailed discussion and an argument for roughly the stance I am taking, see Egan (1995).

[15] Although coherence is something of a minimal requirement for an interpretation to be principled, it should suffice for current purposes. For a more detailed discussion of the issues which arise under the heading of interpretation, see Haugeland (1985: pp. 93-112) or Cummins (1989: pp. 102-113).

Consider the maze illustrated below in figure 2-2. One could imagine a Turing machine which might be interpreted in a manner such that, when provided with strings of tokens which described a location within the maze, would output another string of tokens which would describe a route from that location to the exit of the maze.

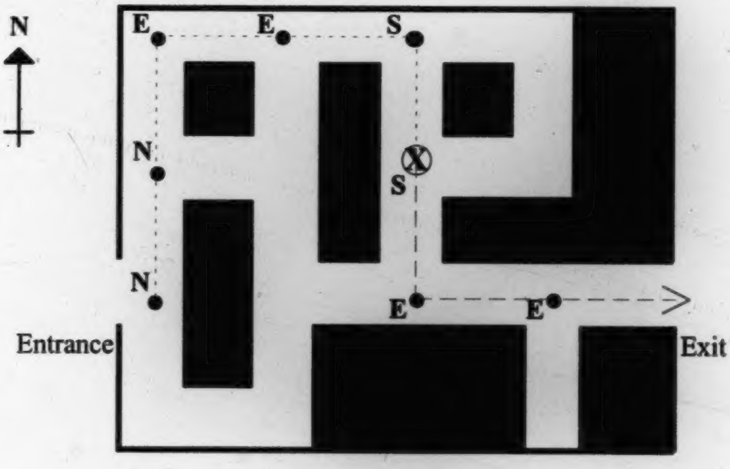

Figure 2-2

---

[16] Even Pylyshyn (1984: pp. 40-48), who is a literalist about cognition being information processing (as opposed to taking this as a metaphor), has some difficulties with the issue of interpretation.

Under these conditions, the initial string of tokens would just be interpreted as 'current location within the maze' and the string of tokens on the tape after the Turing machine had finished all its operations would be interpreted as 'route out of the maze from current location'.

It is important to emphasize that the interpretation, as described, would be *imposed* upon Turing machines. In an example such as this, the interpretation is not something which are intrinsic to Turing machines itself (although the capacity to support interpretations may well be). A consequence of this is that, subject to certain conditions (such as the coherence criterion), there is considerable latitude in possible interpretations of token strings of Turing machines.

Suppose for example a particular Turing machine had operations defined for 4 tokens - 'N', 'S', 'E', and 'W' in its machine table. Suppose further that the Turing machine was set up such that it computed the maze problem, in a manner such that each token could be interpreted as being a direction of movement from a branching point in the maze. So, for example, under this interpretation the position **X** could be specified by the string of tokens 'N N E E S'. If this string of tokens was to be the input to such a Turing machine, the output would be 'S E E'.

In this case, the interpretation takes place at the level of individual tokens. Some theorists (for example Haugeland 1985: p. 91) have assumed that interpretation must take place at this level, however there is no reason why this has to be the case. Indeed, it is more common for interpretations to be specified over complex strings of tokens. The point here is that there is a *choice* about the level at which the interpretation is specified.[17] It is certainly not essential that an interpretation be specified for each of the individual symbols for which operations are defined within the machine table.[18] All that really matters is that strings of tokens can be provided with principled interpretations. However, once this is established, there is nothing to prevent an interpretation of strings of tokens which maps (subject to the coherence criterion) strings of tokens onto mental states. It is the interpretability of Turing machine tokens then (in addition to properties (5) and (6)), which provides the link between the formality condition for mental states and the strings of tokens on a Turing machine tape, under the appropriate circumstances.

The seven properties outlined here lie at the heart of the metaphor which is made between Turing machines (or other computational devices) and the mind (or cognition) and form the core of the CCTM. The CCTM supposes, for the reasons described, that because computational devices such as Turing machines have these properties, then so too do minds. Of course, the analogy between minds and machines if far from perfect, or precise. There is a degree of slippage (as I illustrated whilst discussing property (2), for example), depending upon precisely the kind of computational device under consideration and the exact properties of minds which are taken to be relevant. The imprecision of the analogy undoubtedly accounts for the variations in the precise formulations of the CCTM which have been proposed. However, if it is correct that these seven properties form the core of the position, then we will be able to use these properties to asses the degree to which allegedly non-traditional counter-proposals to the CCTM really differ from the position, and in which respects. In particular, the properties of 'systematicity' and 'compositionality', which arise because of these seven properties, will turn out to be crucial to assessing and evaluating proposed alternatives to the CCTM. However, discussion of these properties will be deferred until later (Chapter VI).

---

[17] Consider the case of the English language. In written form, English is just presented as a string of tokens. However, the interpretation of these strings of tokens occurs at the level of individual words. In principle (provided that the coherence criterion could be satisfied), there is no reason why an interpretation could not be specified at the level of the individual letter, under the appropriate conditions. The situation is similar with the tokens on a Turing machine tape.

[18] Indeed, there are instances of Turing machines where doing this may present very real difficulties. Consider the case of a Turing machine with multiple heads and tapes, discussed by Hopcroft and Ullman (1979: pp. 161-163), or Turing machines with multiple tracks upon its tape (Hopcroft and Ullman 1979: pp. 154-155).

One significant alleged alternative to the CCTM is called 'connectionism'. The consideration of connectionist systems in this respect will be the main focus of this dissertation. The purpose of the next chapter then, will be to introduce connectionist systems, so as to begin the process of assessing the ways in which such systems differ from those presupposed by the CCTM.

III

What is Connectionism?

Introduction

In the previous chapter I introduced the Classical Computational Theory of Mind (CCTM). Although this position is well established in the philosophical literature, it has not been without its critics. Philosophers, such as Searle (1980) and Dreyfus (1991), have argued on a number of grounds that this position is deeply flawed. I want to begin here by briefly discussing some of these objections. These discussions will provide a natural bridge between the material discussed in the last chapter and the main topic of this chapter, connectionist systems.

Philosophical Objections to The Classical Computational Theory of Mind

Probably the earliest well known criticism of the CCTM came from Hubert Dreyfus. In 1965 Dreyfus wrote in a report for the RAND Corporation that "...work in AI resembled alchemy more than science..." (Dreyfus, 1991: p. 2). Right up to the current day, Dreyfus has remained a trenchant critic of the research program. His four best known objections which appear in their most recent form in his 1991 book, *What Computers* Still *Cannot Do*.[1]

Dreyfus argues that there are four assumptions which are foundational to the artificial Intelligence research program. As at least three of these assumptions are shared with (or have close parallels to aspects of) the CCTM, they are important in the current context, because Dreyfus argues that all of these assumptions are false. Dreyfus (1991: p. 156) describes the four objectionable assumptions as follows;

---

[1] This book is the third edition of Dreyfus' *What Computers Cannot Do*, which was originally published in 1972. The third edition contains a substantial volume of material not included in earlier editions. All references to this work will be to this latest version.

> ...the assumption that man functions like a general-purpose symbol-manipulating device [i.e. like a Turing machine] amounts to
>
> 1. A biological assumption that on some level of operation--usually supposed to be that of the neurons--the brain processes information in discrete operations by way of some biological equivalent of on/off switches.
> 2. A psychological assumption that the mind can be viewed as a device operating on bits of information according to formal rules....
> 3. An epistemological assumption that all knowledge can be formalized,...
> 4. ...[T]he ontological assumption that what there is, is a set of facts each logically independent of all others.

Strictly speaking, the biological assumption amounts to nothing more than an empirically testable hypothesis about brain function. As such, this assumption is of no direct relevance to the CCTM. For this reason I will not consider it here.[2] The other three assumptions on Dreyfus' list though are of more import.

The psychological assumption is clearly relevant to the CCTM, as Turing machine tables are just made up of formal rules which operate upon the 'bits of information' encoded upon the machine's tape (Cf. properties (1), (2), (3), (4) and (6) discussed in the previous chapter). According to Dreyfus though this assumption should also be rejected. Dreyfus has no objection, in principle, to the psychological assumption as a hypothesis. What he takes to be problematic is when the assumption is adopted *a priori*. Unfortunately, he believes, these two statuses of the assumption have been confused in the literature. If the psychological assumption is treated as an empirically testable hypothesis, then it runs into problems due to the fact that the empirical results of computational simulations are, in Dreyfus' (1991: p. 187) opinion

---

[2] Dreyfus believes this assumption is false, on the grounds that it is not supported by the neurological evidence. Of course there is always the possibility that as research into the brain advances, this could change.

> ...riddled with unexplained exceptions, and unable to simulate higher-order processes such as zeroing in and essential/inessential discrimination....

In contrast, the psychological assumption is untenable as an *a priori* truth, as there are no valid *a priori* arguments which can be given for the assumption. Thus, Dreyfus urges that the psychological assumption cannot be justified and consequently should be abandoned.

Crucial to Dreyfus' objection here is his assessment of the success of the CCTM inspired research program. Most cognitive scientists would not even attempt to argue that this assumption is an *a priori* truth, but rather take it as a working hypothesis, which may or may not prove to be fruitful. However, assessments such as Dreyfus' here are notoriously unreliable. One need only recall Kant's assessment that logic was a completed science, or the case of the U.S. President who wanted to shut down the patent office, on the grounds that he believed that everything which could be invented already had been, to realize the dangers inherent in such assessments. The other significant point here is that Dreyfus' assessment is defeasible. As such, his objection does not provide a knock down argument against the continuation of a research program based upon the CCTM. Thus his conclusion, that research programs, such as that based upon the CCTM, which embrace the psychological assumption should be abandoned, does not necessarily follow. The alternative conclusion, that such research programs should be pursued with greater vigor, is also, strictly speaking, compatible with Dreyfus' premises.

The link between what Dreyfus terms the 'epistemological assumption' and the CCTM is, perhaps, not immediately apparent. However, the connection becomes more obvious when Dreyfus (1991: p. 190) suggests that the assumption actually involves the two sub-claims,

> (a) that all nonarbitary behavior can be formalized, and
> (b) that the formalism can be used to reproduce the behavior in question.

There are many ways in which the epistemological assumption, seen in this light, is similar to the psychological assumption, though weaker. Whereas those who accept the psychological assumption suppose that *the very same rules* are used by biological cognitive agents as those used in the formalization of behavior, those who accept just the epistemological assumption are only committed to the claim that there is *some set of rules* which can be used to formalize behavior and which are sufficient to produce the behavior in question. The link between this assumption and the CCTM is consequently in many ways similar to that between the CCTM and the psychological assumption. [3]

Dreyfus argues against both of the constitutive sub-claims of the epistemological assumption. He suggests that sub-claim (a) amounts to an unjustified generalization from physical science. Sub-claim (b), he believes, is not only false, but also untenable. This is because, on Dreyfus' analysis, the behavior of natural cognitive systems is such that formalisms of the relevant type cannot reproduce the required behaviors, as to do so would require a non-terminating regress of rules. As a consequence, Dreyfus urges the rejection of the epistemological assumption too.

---

[3] It is instructive to compare Dreyfus' distinction between the two assumptions and Pylyshyn's (1984) distinction between strong and weakly equivalent systems.

The ontological assumption is fundamental to research undertaken within the scope of the CCTM because, in Dreyfus' (1990: p. 206) words,

> ...the data with which the computer must operate...must be discrete, explicit, and determinate; otherwise, it will not be the sort of information which can be given to the computer so as to be processed by rule.

The tokens of Turing machines (the 'data' upon which they operate) have all these properties (Cf. properties (1), (2), (3) of the previous chapter).

Dreyfus (1990: pp. 211-212) believes that one reason that the ontological assumption is so readily accepted, is because this idea has a very long philosophical tradition. However, he goes on to argue that the ontological assumption is false. In making this argument he cites the difficulties which computational systems have run into when dealing with very large databases and the so-called 'problem of commonsense'[4] as evidence for the falsehood of the assumption. In Dreyfus' view, there just is no logically independent list of objects and facts about each object, and consequently the ontological assumption, like the other assumptions, should be rejected.

The importance of Dreyfus' work in the current context lies not so much in his objections to aspects of the CCTM, but rather in the strategy he seems to favor to avoid these objections. Dreyfus believes that the way to avoid the problematic assumptions is to employ a connectionist approach to understanding the mind. In his (1991: p. xiv) opinion,

> ...the neural-network modelers had a much more plausible answer to the question, If not symbols and rules, what else?

---

[4] See Baumgartner and Payr (1995: pp. 17-18) for a brief description of this problem.

Dreyfus is none the less cautious about networks. Although he sees a great deal of potential in the network based approach, he (1991: pp. xv-xvi) maintains that it has yet to be seen whether the full potential will be actualized. Nonetheless, in a recent interview (Baumgartner and Payr 1995: p. 82) he made the following prediction;

> I predict that within ten years there won't be any cognitivism [Dreyfus' term for the traditional view of the CCTM] or symbolic Artificial Intelligence around and the researchers will have turned to neural network simulation...

Dreyfus is not alone in believing that network models provide a radical, and perhaps more tenable, way of understanding the mind than the CCTM. In recent years, a number of philosophers have thrown their support behind the connectionist research program for a variety of reasons. Amongst these philosophers is the other trenchant critic of the computational theory of mind, John Searle. Searle (1992: pp. 246-247) defends connectionism (albeit in a limited way) as an alternative to the CCTM. He claims that,

> Amongst their other merits, at least some connectionist models show how a system might convert a meaningful input into a meaningful output without any rules, principles, inferences, or other sorts of meaningful phenomena in between. This is not to say that existing connectionist models are correct--perhaps they are all wrong. But it is to say that they are not all obviously false or incoherent in the way that the traditional cognitivist models...are.[5]

So, if Searle and Dreyfus are correct, then network based (or 'connectionist') models provide a basis for an alternative to the traditional CCTM. However, what is less clear is exactly in which respects connectionist models provide such an alternative. Is it the case that network models have *none* of the properties associated with the CCTM, as discussed in the last chapter, or do they only differ with respect to *some* properties? If the latter is the case, then which properties do network models have in common with the CCTM? To begin to answer such questions, I will first describe the features of one class of network models.

---

[5] In a recent interview (Baumgartner and Payr, 1995: pp. 203-213) Searle also affirms his optimism for the network research program.

What Is Connectionism?

Connectionism is a style of modeling based upon networks of interconnected simple processing devices. This style of modeling goes by a number of other names too. Connectionist models are also sometimes referred to as 'Parallel Distributed Processing' (or PDP for short) models or networks.[6] Connectionist systems are also sometimes referred to as 'neural networks' (abbreviated to NNs) or 'artificial neural networks' (abbreviated to ANNs). Although there may be some rhetorical appeal to this neural nomenclature, it is in fact misleading as connectionist networks are commonly significantly dissimilar to neurological systems. For this reason, I will avoid using this terminology, other than in direct quotations. Instead, I will follow the practice I have adopted above and use 'connectionist' as my primary term for systems of this kind.

The basic components of a connectionist system are as follows;

1) A set of processing units
2) A set of modifiable connections between units
3) A learning procedure (optional)

---

[6] Although in current usage, the terms 'connectionist' and 'PDP' have effectively become synonyms, the two terms once had different meanings. Originally, so-called 'Connectionist' models were generally associated with Ballard's work at the University of Rochester. So-called 'PDP' models, on the other hand, were associated with the PDP Research Group of San Diego (for more details on the etymology of these terms, see Smolensky 1991: p. 225, fn. 5). I will follow what is now current practice and use the two terms as synonyms.

I will describe each of these components in turn. Readers who require further technical details should consult the general framework for connectionist systems described by Rumelhart, Hinton and McClelland (1987).

Processing Units

Processing units are the basic building blocks from which connectionist systems are constructed. These units are responsible for performing the processing which goes on within a connectionist network. The precise details of the processing which goes on within a particular unit depends upon the functional subcomponents of the unit. There are three crucial subcomponents. These are,

a) The net[7] input function
b) The activation function
c) The output function

The various components of a processing unit can be represented as follows,

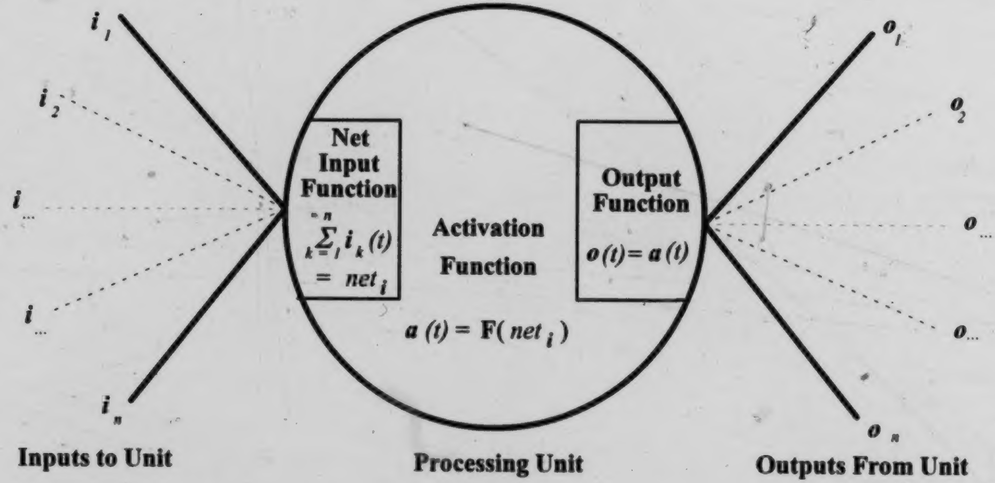
i_1
i_2
$i_{...}$
$i_{...}$
i_n
Net Input Function
$\sum_{k=1}^{n} i_k(t) = net_i$
Activation Function
$a(t) = F(net_i)$
Output Function
$o(t) = a(t)$
o_1
o_2
$o_{...}$
$o_{...}$
o_n
Inputs to Unit
Processing Unit
Outputs From Unit

Figure 3-1

---

[7] The term 'net' here is not meant as an abbreviation of the term 'network'. The intended sense is that of 'net' as opposed to 'gross'.

The *net input function* of a processing unit determines the total signal that a particular unit receives as a function of all the inputs to the network at time *t*. The net input function takes as input the signal which a unit receives from all sources (i_{i-m}), including the other units which it is connected to. It is often the case that the net input function of a unit is relatively simple. Commonly, the net input function for a unit will just sum the of the input signals the unit receives at a particular time (*t*).

The *activation function* of a particular unit determines the internal activity of the unit, depending upon the net input (as determined by the net input function) that the unit receives. There are many different kinds of activation functions which particular units can employ. The 'type' of a particular unit is determined by its activation function. Perhaps the simplest kind of activation function is illustrated below,

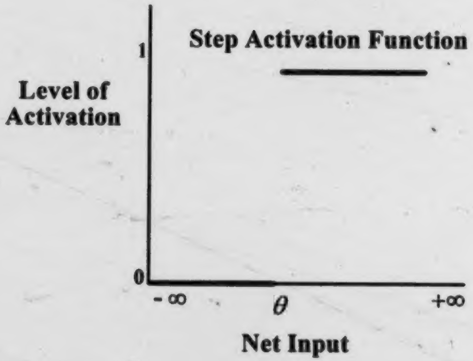

Figure 3-2

Activation functions such as this act rather like switches and are sometimes called 'step functions'. If the net input to a unit employing such an activation function is greater than some threshold value, θ, the unit becomes fully active.[8] If the net input is below this level, the processing unit is totally inactive. The activation function, $\boldsymbol{a}_j$, for such a unit, j, can be expressed more formally as follows;

$$\boldsymbol{a}_j = \begin{cases} 0 \text{ if } \boldsymbol{i}_j < \theta_j \\ 1 \text{ if } \boldsymbol{i}_j > \theta_j \end{cases}$$

where $\boldsymbol{i}_j$ is the net input received by the unit at time t and θ_j is the threshold value for unit j.

Activation functions of this kind were used in the very earliest days of network research. Unfortunately though they are subject to certain significant limitations (see Minsky & Papert 1968). In particular, it is not possible to train networks which employ this kind of unit arranged into more than two layers.

Currently, within the domain of trainable networks, by far the most common kind of processing unit employed by connectionists is what Ballard (1986) has called an 'integration device'. The logistic function described by Rumelhart *et al* (1986a: pp. 324-325), for example, is an instance of an integration device. Integration devices have a sigmoidal activation function, similar to the one illustrated below, and can be described as a continuous approximation of a step function.

---

[8] Note, the activation levels need not be 0 and 1. These values are employed merely for illustrative purposes.

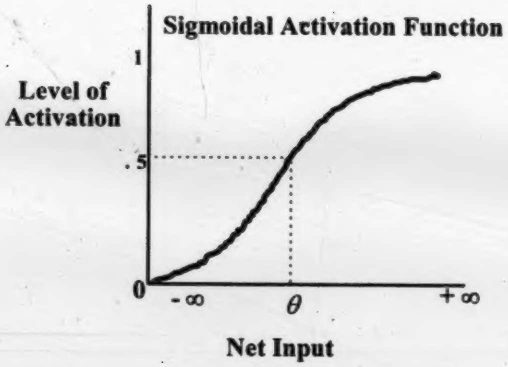

Figure 3-3

The activation function, a_j, for a unit, *j*, of this variety, receiving net input i_j is;

$$a_j = \frac{1}{1 + e^{-i_j + \theta_j}}$$

Integration devices include in their activation function something known as 'bias'. Bias serves to alter the level of input to a unit which is needed for that unit to become active and is therefore analogous to the threshold of a step function. In more technical terms, bias serves to translate the activation function along an axis representing net input, thereby altering the location of the activation function in net input space. The θ_j term in the logistic equation is the bias term of that activation function.

One important feature of sigmoidal activation functions is that they be differentiable. The reason this is important is that it make it possible to train networks with more than two layers of processing units, using powerful learning rules such as the generalized delta rule, described by Rumelhart, Hinton and Williams (1986a: pp. 322-328). This ability to train networks with multiple layers has greatly increased the power of networks.

Although integration device units are arguably the most commonly employed unit type in trainable networks at the current time, other activation functions have also been explored. Recently, Dawson and Schopflocher (1992) have described a kind of processing unit which they call, following Ballard's (1986) terminology, a 'value unit' . Value units employ a Gaussian activation function, such as the one below,

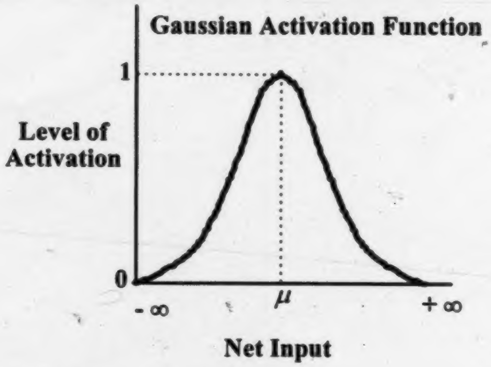

Figure 3-4

The activation function, a_j, for a unit, j, of this variety, receiving net input i_j is;

$$a_j = exp\left[-\pi\left(\mathbf{i}_j - \mu_j\right)^2\right]$$

As the net input, i_j, to a value unit increases, the level of activation of the unit, a_j, increases, but only up to a certain point, μ_j. When $i_j = \mu_j$, the activation a_j is maximized and has a value of 1. If the unit receives net input greater than μ_j, the activation of the unit begins to decline again, down to 0. As a consequence of having this kind of activation function, value units will only generate strong activation for a narrow range of net inputs. Value units, like integration devices, can be used to construct trainable multilayered networks.

A unit in a connectionist network typically sends a signal to other units in the network or to outside the network. The signal that a unit sends out is determined by the *output function*. The output function depends upon the state of activation of the unit. It is common practice, at the current time, that the output function of a particular unit is such that it just sends out a signal equivalent to its activation value. However, there is no theoretical reason why this must necessarily be the case.

Modifiable Connections

In order for a particular connectionist network to process information, the units within the network need to be connected together. It is via these connections that the units communicate with one another. The connections within a network are usually 'weighted'. The weight of a connection determines the amount of the signal input into the connection which will be passed between units. Connection weights (sometimes also called 'connection strengths') are positive or negative real numerical values. The amount of input a particular connection supplies to a unit to which it is connected is the value of the result of the output function of the sending unit, multiplied by the weight of the connection.

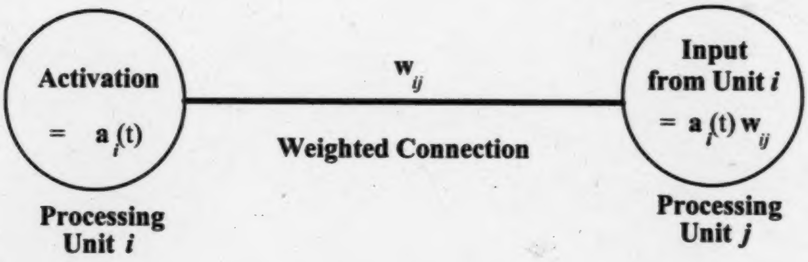

Figure 3-5

In principle, there is no limit to the number or pattern of connections which a particular unit may have. Units can have weighted connections with themselves and there can even be loops or cycles of connections. However, for current purposes there is no need to explore such complexities. Instead, attention will be limited to simple three layered systems like the one illustrated below.

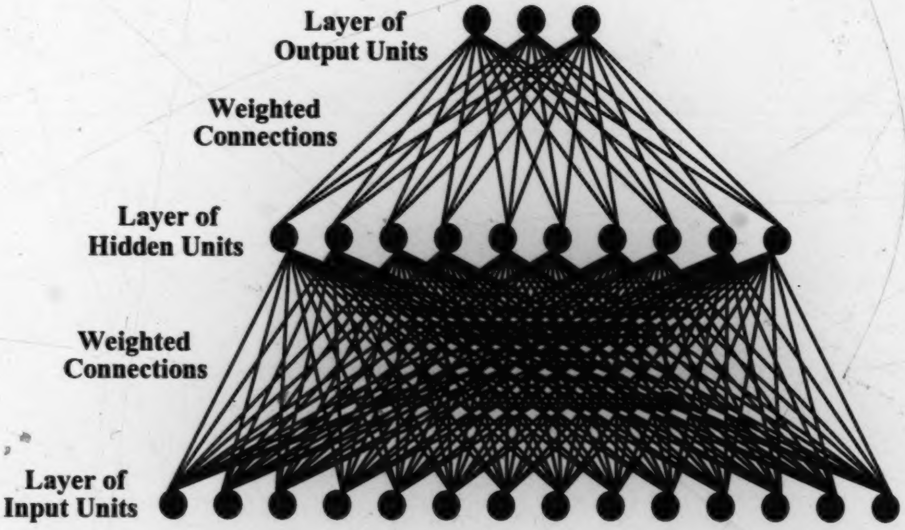

Figure 3-6

If particular processing units within a system can receive inputs from sources external to the network itself, then these units are usually called *input units*. Alternatively, if particular processing units can send signals outside the network itself, then these units are usually called *output units*. Finally, processing units which can only directly communicate with other units within the network (i.e. units which have no direct inputs or outputs which are external to the network) are usually called *hidden units*. Layers of hidden units are not an essential feature of networks, although many networks require a single layer of hidden units to solve particular problems. It is also the case that there is no reason why a network should just have a single layer of hidden units. For example, a network described by Bechtel and Abrahamsen (1991: p. 169) has two layers of hidden units.

Learning Rules

A learning rule is an algorithm which can be used to make changes to strengths of the weights of the connections between processing units. Whereas all connectionist systems have processing units and patterns of connections between the units, not all systems have learning rules. Some networks (e.g. the Jets and Sharks Interactive Activation and Competition network, described in McClelland and Rumelhart (1988)) are built by hand (or 'hand-coded'). Hand-coded networks have the weights of the connections between the processing units set manually by the network's builder. However, in most connectionist networks a learning rule of some kind is employed. In this dissertation I will be concerned primarily with networks that employ learning rules.

A learning rule is used to modify the connection weights of a network so as (hopefully) to make the network better able to produce the appropriate response for a given set of inputs. Networks which use learning rules have to undergo training, in order for the learning rule to have an opportunity to set the connection weights. Training usually consists of the network being presented with patterns which represent the input stimuli at their input layer. It is common for connection weights to be set randomly prior to training.

For example, consider one of the most popular learning rules for connectionist networks, Rumelhart, Hinton and McClelland's (1986) generalized delta rule. When using this rule, the network is shown example patterns from a training set. The purpose of the generalized delta rule is to modify the network's connection weights in such a way that the network generates a desired response to each pattern in the training set.

More specifically, with the generalized delta rule learning proceeds by presenting one of the patterns from the training set to the network's input layer. This causes a signal to be sent to the hidden layer(s), which in turn results in a signal being sent to the output layer. In the generalized delta rule, the actual activation values of each output unit are compared to the activation values that are desired for the input pattern. The error for each output unit is the difference between its actual and desired activation. The generalized delta rule uses this error term to modify the weights of the connections that are directly attached to the output units. Error is then sent through these modified weights as a signal to the hidden units, which use this signal to compute their own error. The error computed at this stage is then used to modify the connection weights between the input units and the hidden units. In every case, when a weight is changed, the generalized delta rule guarantees that this change will reduce the network's error to the current input pattern.

Usually, the learning rule only makes small changes to the connections weights between the layers each time it is applied. As a result training often requires numerous presentations of the set of input patterns. By the repeated presentation of the training set and application of the learning rule, networks can learn to produce the correct responses to the set of inputs which make up the training set. Learning rules thus offer a means of producing networks with input/output mappings appropriate to particular tasks or problems. Each presentation of the set of input patterns and output patterns is known as an 'epoch' or a 'sweep'. When the network produces an output for each input pattern which is close enough (as determined by the experimenter) to the desired output for each pattern, training stops and the network is said to have 'converged'.

Conclusion

In the previous chapter the CCTM, which is based upon devices such as Turing machines, was described. Intuitively, it seems obvious that connectionist networks, as described here, are very different from Turing machines. For example, whereas the heads of Turing machines process tokens on the tape one at a time, connectionist systems employ many simple processors which can operate in parallel. Similarly, the tokens upon which a Turing machine operates are located upon the machine's tape and the operations are specified separately in the machine table, yet there does not seem to be any such obvious analogous distinction in the case of connectionist networks. Reasons such as these, provide some *prima facie* plausibility to the claim that connectionist networks might provide the basis for an alternative conception, or interpretation, of the CCTM.

Indeed, a number of significant and strong claims have been made about connectionist systems, with respect to the CCTM. For example, Schneider (1987) has argued that connectionist research represents a 'paradigm shift' (in the Kuhnian sense) away from the CCTM for psychology. Similarly, Smolensky (1988: p. 3) has even claimed that,

> [Connectionist models] may possibly even challenge the strong construal of Church's Thesis[9] as the claim that the class of well-defined computations is exhausted by those of Turing machines.

---

[9] This is the same thesis that I have been referring to as the 'Church-Turing Thesis'.

Moreover, strong philosophical claims have been advanced, based upon connectionist models. Smolensky (1988: p. 3) again provides us with a good example when he claims that,

> It is likely that connectionist models will offer the most significant progress of the past several millennia on the mind/body problem.

Churchland P. M. (1990: p. 165) believes that similarities between the brain and connectionist systems are sufficient for networks to "...give some real substance..." to the eliminative materialist position in the philosophy of mind (see also Churchland P. M. 1989).

In the philosophical literature, connectionist models have been claimed to have particular significance for representational issues. Clark (1989: p. 124) maintains that,

> ...the use of a PDP architecture opens up new and qualitatively different avenues of searches and representations to those so far explored in conventional AI.

Sterelny (1990: p. 168) introduces connectionism (although he ultimately rejects the view) by noting that,

> Connectionists offer a rival view [to the CCTM] of the architecture of the mind, the nature of mental representation, and the nature of operations on those representations.

In the context of his discussion of the CCTM, Cummins (1989: p. 157, fn. 6) highlights the difference between the CCTM and connectionist inspired theories when he notes that,

> Connectionists do not assume that the objects of computation are objects of semantic interpretation.

In the light of claims such as these, it is small wonder that both Dreyfus and Searle view connectionism as a potential source of an alternative to the CCTM!

However, there are reasons to be cautious about accepting connectionism as a dramatic alternative to the CCTM. In the next chapter, I will describe in more detail arguments for distinguishing connectionism from the CCTM. But, I will also show that these arguments are less impressive than they initially appear. Understanding the limitations on the claims which have been made about connectionist systems is especially important for philosophers. This is because philosophers are amongst those who are most likely to be mislead by these claims, as by and large philosophers lack the technical background to correctly disambiguate the claims. As a consequence, philosophers are most at risk of taking these claims to be true in cases where they are not, and as a result inadvertently including such claims as premises in their arguments.

IV
The Myths of Connectionism

Introduction

In the previous chapter, connectionist systems were introduced. Although the idea that networks might offer an alternative to the traditional CTM has some *prima facie* plausibility from the nature of such networks themselves and some broad philosophical support, the exact reasons why networks might be thought to challenge the traditional CTM have not been examined in detail

There are a number of reasons which have been given in the literature for the contention that connectionist systems are not only substantially different from the kinds of systems which are usually associated with the traditional CTM, but also, in a significant sense, offer a better framework within which to model aspects of cognitive functioning. These reasons, as often as not, depend upon claims about the properties of connectionist systems. Unfortunately, it is often the case that the claims upon which these proffered reasons rest do not stand up to close critical scrutiny and, at the very least, require very careful qualification. These claims form what I call 'The Myths of Connectionism'. The goal of this chapter is to discuss several of the more significant and commonly encountered connectionist myths, so as to clear the way for a more careful assessment of the relationship between connectionist systems and the traditional CTM.

Before proceeding to the discussion of the first of these myths, it is worthwhile introducing the term 'GOFAI'. 'GOFAI' is term coined by Haugeland (1985: p. 112), which is short for '*G*ood *O*ld *F*ashioned *A*rtificial *I*ntelligence'. GOFAI systems are, according to Hagueland's usage of the term, systems based on the traditional conception of the CTM (See Chapter 2 for details). The purpose of introducing this term is that it provides a useful shorthand for the discussion of the relationship between connectionist systems and systems of this (allegedly different) type.

Myth 1: 'Connectionist systems are biologically plausible'.

One important series of related claims made by advocates of the connectionist approach (such as McClelland, Rumelhart and Hinton (1987), Rumelhart (1989), for example) is that, in a significant sense, connectionist architectures are considerably more brain-like because they are more biologically plausible than GOFAI architectures. This myth is often appealed to by both philosophical friends and foes of connectionism. For example Sterelny (1990: p. 175) and Cummins (1989: p. 155), though neither of them are great fans of the connectionist approach, both appeal to this myth.

It is the advocates of the connectionist approach though, who most frequently appeal to this myth. Clark (1989: p. 4), for example, talks of "...the brain-like structure of connectionist architectures." Similarly, Bechtel and Abrahamsen (1991: p. 17), in describing the rise of the new connectionism, claim that "...network models were attractive because they provided a neural-like architecture for cognitive modeling". Perhaps the most explicit endorsement of this myth though, is due to Paul Churchland. Churchland (1989: p. 160) introduces connectionist networks as follows,

> The networks to be explored attempt to simulate natural neurons with artificial units...Each unit receives input signals from other units via "synaptic" connections...the "axonal" end branches from other units all make connections directly to the "cell body" of the receiving unit.

Churchland makes similar claims elsewhere too (see Churchland 1988: p. 156, for example). Even Dennett (1991: p. 239) makes reference to "...'connectionist' architectures of neuron-like elements...". Given these examples, I hope it is clear that this claim is well established in the philosophical literature.[1]

There are, in fact, two particular related claims which are frequently confused in the literature. These are,

(a) Connectionist systems are biologically plausible, and

(b) Connectionist systems are more biologically plausible than GOFAI architectures.

Although there may be some credibility to the second claim (b), it is the first claim (a) which is, unequivocally, a myth. A careful comparison of the various components of a connectionist system with the supposedly analogous components of the brain shows that there is only the most minimal similarity between the biological and connectionist systems. As, to some degree, claim (b) rests upon claim (a), these facts cast some doubt on the plausibility of this claim too.

Processing Units

Let us begin by examining the claim that a connectionist processing unit is in some sense similar to a biological neuron. For example, Rumelhart (1989: p. 134) has claimed that a "...[connectionist] processing unit [is] something close to an abstract neuron." This claim should arouse immediate suspicion, given the fact that, as Winlow (1990a: p. 1) notes "It has always been very clear to neuroscientists that there is no such thing as a typical neurone,...". There are, as a matter of fact many different types of neurons (see Kolb and Whishaw (1990: p. 5) and deGroot and Chusid (1988: p. 5) for illustrations of some of these types). Indeed, according to Churchland and Sejnowski (1994: pp. 43) there are twelve different kinds of neurons in the neocortex alone. Given these facts, it seems reasonable to ask, just which *kind* of neuron connectionist processing units are an abstraction from. Connectionists though, as a rule, have little if anything to say on this matter.

---

[1] One immediate concern which might arise over the claims about the relationship between connectionist networks is the question of why the biological plausibility (or otherwise) of particular systems, is relevant to assessing the appropriateness of a particular class of systems as models of cognitive function. After all, for the traditional CTM, when wedded to functionalism, this is not an issue (for a discussion of the reasons why this is the case, see Sterelny 1990: pp. 1-6). I take it that this concern with what Pylyshyn (1984) would call the 'implementational level' is a symptom of connectionists wishing to distance themselves from standard positions.

If the 'abstract neurons' employed within connectionist systems are supposed to capture the significant features of the class of *all* neurons, then it is reasonable to ask how the set of features selected were decided upon. Regrettably though, the selection of features and functional properties employed in 'abstract neurons' has yet to be justified or defended in any detail. Thus, until some better account of the relationship between connectionist processing units and actual biological neurons is forthcoming, it seems reasonable to treat this claim about processing units with some skepticism. A bold unsubstantiated claim will not suffice, where argument is required.

A related concern derives from the fact that many, if not most, connectionist systems involve homogeneous processing units.[2] This homogeneity does not reflect the complexity of the biological situation. Getting (1989: p. 187) remarks that "No longer can [biological] neural networks be viewed as the interconnection of many like elements....".

---

[2] A notable exception to this general rule can be found in Dawson and Schopflocher (1992: pp. 25-26).

In fact, Churchland and Sejnowski (1994: p. 51) claim that, in the brain "[m]ost connections are between, not within, cell classes." If connectionist networks were to be really biologically plausible, it is reasonable to expect them to reflect these facts about biological systems. The discrepancy between the state of affairs in connectionist networks as compared to biological neural networks merely serves to undermine the tenability of the claim that connectionist systems are biologically plausible.

Finally, it is common practice for connectionist models (which undergo training employing a learning rule) to have the bias trained at the same time as their connection weights are trained. However, there is little or no evidence that threshold membrane potentials (the most natural biological equivalents of bias) in biological systems can be modified in any analogous way. In natural neurons, there is no evidence that the thresholds of neurons exhibit any plasticity at all (for more details, see Dawson and Shamanski (1993)). This shows that connectionists, whose systems involve trainable biases, standardly take it upon themselves to add an extra degree of freedom into their networks. However, this degree of freedom lacks any biological justification. Again, this counts against the biological plausibility of such systems.

Connections

In biological nervous systems, neurons have two components which are roughly equivalent to the weighted connections between processing units in a connectionist network. These components are axons and dendrites. Dendrites receive signals into the neuron and axons send signals from particular neurons to others. One immediate (though relatively trivial) difference between connectionist systems and biological ones is that, in biological systems, axons and dendrites are components of neurons themselves, whereas in connectionist systems the connections between units are distinct from the units themselves. This however, is not the only difference.

It is standard practice for connectionists to make their networks 'massively parallel'. That is to say, each unit of a particular layer is normally arranged so that it has connections to every unit of both prior and subsequent layers in the network (even if these connections are zero-weighted). However, there are no results which suggest that this is the situation in biological systems (see Dawson and Shamanski 1993). Indeed, what evidence there is suggests that this is not the case. Churchland and Sejnowski (1994: p. 51), whilst discussing the patterns of connectivity found in the brain cortex note that,

> Not everything is connected to everything else. Each cortical neuron is connected to a roughly constant number of neurons, irrespective of brain size, namely about 3% of the neurons underlying the surrounding square millimeter of cortex. Hence,...cortical neurons are actually rather sparsely connected...

It is also the case that in standard small connectionist networks individual units from one layer can have a significant impact on the activation level of particular units at the next layer. In biological systems though, the influence of one neuron upon the state of another is, in most cases (there are important exceptions), relatively weak. Usually, the influence of one neurons activity upon another is in the order of 1%-5% of the firing threshold (see Churchland and Sejnowski 1994: p. 52). In the connectionist literature, no attention is paid to this particular subtlety.

Another sharp discrepancy which exists between standard connectionist models and biological systems is in their differing ways of transmitting signals between units or neurons. In connectionist networks, the signals which are sent via the weighted connections take the form of continuous numerical values. But in real neurological systems, signals are sent in the form of spiked pulses of signal (for an illustration of this, see Churchland and Sejnowski (1994: p. 53)). This would not be a decisive objection against connectionist models, were it to be the case that continuous values could capture the essential properties of the signals transmitted by the spiked pulses. However, this is not the case. Firstly, different types of neurons have different firing patterns. Secondly, some neurons firing patterns are a function of their recent firing history. Thirdly, some neurons have oscillatory firing patterns. Fourthly, most neurons spike randomly, even in the absence of input (Churchland and Sejnowski 1994: pp. 52-52). Finally, it is also the case that signals between neurons in biological systems are sent by more than one medium. Synaptic transmission occurs by both electrical and chemical means (Getting 1989: p. 191). Although it may be possible to capture at least some aspects of these complexities with the continuous values standardly employed in connectionist networks, there is no reason to believe that this is entirely the case without an argument to this effect. Connectionists have yet to come up with such an argument. Indeed, there seem to be good grounds to believe that the properties just mentioned will be highly significant to the functioning of actual neural systems. This being the case, there seem to be good grounds for doubting the putative biological plausibility of connectionist networks.

In most connectionist networks, the relationship between the signal sent down a connection and the influence of that connection (with the associated weighting) upon the receiving unit is fairly straightforward. This is not the case in real neural systems though. Dreyfus (1993: pp. 161-162) briefly describes work by Lettvin which suggests that axon branches may serve to act as "low pass filters with different cutoff frequencies", with the precise frequency being dependent upon the physical diameter of the actual axon branch. This being the case, there will be a complex and functionally significant relationship between the frequency and pattern of neuronal firing, and the length and diameter of the connections between neurons. This relationship will be functionally significant as it will have a direct effect upon the influence of one neuron upon another. However, there is nothing in standardly described connectionist systems which is even remotely similar to such a mechanism. This being the case, there must be at least some functionally significant properties of biological systems which are not captured in connectionist systems. This, again, mitigates against the tenability of connectionist claims to biological plausibility.

Hopefully the facts from neuroscience cited above are sufficient to show that connectionist claims to biological plausibility are not as straightforward as many of the proponents of the myth would have us believe. Indeed, there are significant functional differences between connectionist systems and biological ones. Given these facts, it seems reasonable to conclude that the claim that connectionist systems are biologically plausible, at the current time at least, is in large part a myth.

As I noted above though, even if the claim that connectionist systems are biologically plausible is not tenable, there is a weaker claim, to the effect that connectionist systems are more biologically plausible than their GOFAI counter-parts. As a matter of fact, it is not too uncommon to find both the stronger and weaker claims being made at the same time in the literature. For example, McClelland, Rumelhart and Hinton (1986: p. 12) seem to be doing just this when they remark that,

> One reason for the appeal of PDP models is their obvious "physiological" flavor: They seem so much more closely tied to the physiology of the brain than other [i.e. GOFAI] kinds of information-processing models.

There is perhaps more plausibility to the weaker claim, although it too has problematic aspects (for example, it is far from clear what the appropriate metric should be for assessing comparative biological plausibility). However, a *prima facie* case for the plausibility of the weaker claim can be made.

Consider the case of two computational systems which both model some cognitive capacity. Let us suppose further that one system is connectionist and the other is a production system (production systems are usually fairly prototypical cases of GOFAI models). If for some reason (perhaps a desire to develop a system which was strongly equivalent to human beings) we wished to try to make each system more biologically plausible, how would we fare?

In the case of the connectionist system, there are a number of steps which might be taken. These range from substituting non-homogeneous processing units with activation functions similar to the biological neurons of the relevant type, to utilizing more complex mechanisms to mediate the transmission of signal between units. Shastri and Ajjanagadde (1993), for example, have described a system which mimics in a rudimentary manner, the spiking of neural firing.[3] How, on the other hand might we go about making a production system more biological? There does not seem to be any straightforward manner of doing this. Adding more productions is very unlikely to do the trick! So, in theory, connectionist systems *could be* made more biologically plausible than their GOFAI cousins. This, though, is not the same as the claim that connectionist systems *actually are* more biologically plausible at the current time. Once again, this claim, if made in the present tense (C.f. the remark made by McClelland, Rumelhart and Hinton (1986: p. 12), cited above), is little more than a myth.[4]

Myth 2: 'Connectionist Systems Are Consistent With Real Time Constraints Upon Processing'

There is another claim which is sometimes made on behalf of connectionist systems, which is based upon comparing them with biological cognitive entities. This claim too has a significant mythological component.

One of the astonishing things about biological cognitive systems is the speed at which they are able to perform tasks which (apparently) require many complex calculations. Somehow or other, the neurological components of humans and animals are able to successfully perceive the world, remember things and so on, despite the fact that individual neurological components (such as neurons) operate slowly, when compared, for example, to the speed of a modern microprocessor. These facts have lead some connectionists (for example, Feldman and Ballard 1982, Bechtel 1985, Rumelhart 1989, Shastri 1991) to argue for the adoption of their approach. Such arguments frequently appeal to the problems which can arise with GOFAI systems, with respect to real time constraints upon processing.

---

[3] Not too much weight should be put on this system though - it is very far from being biologically plausible. See Dawson and Berkeley (1993).

[4] For a further discussion of the claim that connectionist networks have some kind of biological plausibility, see Quinlan (1991: pp. 240-244). Quinlan's assessment of the current state of the art is similar to mine.

One of the best known versions of this type of argument is the so called "100 step" argument (This nomenclature originates from Feldman and Ballard 1982). It is argued that, from what is known about the speed of firing of neurons in the brain, many basic human cognitive capacities (those which take under a second for humans to process in real time) *cannot* involve more than about one hundred processing steps. This is because actual neurons cannot go through more than about one hundred states in under a second. As standard GOFAI architectures are serial in nature and usually require considerably in excess of one hundred steps, connectionists argue that they cannot provide a good model of actual cognitive function.

Rumelhart's (1989: p. 135) version of the argument goes like this;

> Neurons operate in the time scale of milliseconds, whereas computer components operate in the time scale of nanoseconds--a factor of 10^6 faster. This means that human processes that take on the order of a second or less can involve only a hundred or so time steps.

Rumelhart then goes on to list several processes which occur in a second or so. The processes listed are all significant for the study of cognition and include linguistic capacities, perception and memory retrieval. The claim is that the facts about the speed of operation of neurons means that realistic computational accounts of these cognitive processes must either involve less than one hundred or so operations, or some account must be given for how it is that more than one hundred operations can occur in less than a second.

Rumelhart (1989: p. 135) believes that the correct way to explain phenomena of this type is a follows:

> Given that the processes we seek to characterize are often quite complex and may involve consideration of large numbers of simultaneous constraints, our algorithms *must* involve considerable parallelism....Although the brain has *slow* components, it has *very many* of them....Rather than organize computation with many, many serial steps, as we do with systems whose steps are very fast [i.e. GOFAI systems], the brain must deploy many, many processing elements cooperatively and in parallel to carry out its activities.

Devices such as a Turing machines or von Neumann machines (usually) have a single processor which performs operations one at a time, one after another. This is often called 'serial' processing. One of the features of connectionist systems, by contrast, is that they are constructed from many simple processing devices which operate at the same time as one another. This is often referred to as 'parallel' processing. The parallel nature of connectionist systems means that they can (theoretically) perform many operations within each time step and thus, it is claimed, they do not (necessarily) violate the 100 step constraint.

The argument sketched above is, pretty clearly, another connectionist argument for the biological plausibility of their systems. It differs from the arguments of the previous section though, in so much as the arguments plausibility depends upon the presumption that there is some functionally significant similarity between connectionist processing units and biological neurons (this point is noted by Fodor and Pylyshyn 1988: p. 55). As we have seen in the previous section though, the claim that connectionist units are neuron-like is largely a myth. This is not the only reason why the claim that connection systems are consistent with real time constraints upon processing is dubious, however.

The entire 100 step argument turns upon the premise that the individual neuron is the computationally significant level, as far as speed constraints go, in the brain. Should it turn out that there is significant processing which occurs at the sub-neuronal level (for example, at the level of synaptic clefts, see Kolb and Whishaw 1990: pp. 46-47) in the brain, then this argument would loose much of its plausibility. The additional processing steps which standard architectures seem to require may be being done at this sub-neuronal level. Furthermore, Fodor and Pylyshyn (1988: p. 55, n31) note that there are many chemical processes of the dendrites of biological brains which take place over a wide range of time scales. This being the case, there are grounds for wondering why advocates of the 100 step argument choose the rate of neuronal firing as the relevant time scale for their argument. There is no defense of this choice in the connectionist literature.

As a matter of fact, what is known about the behavior of biological systems tends to make the appeal to the speed of neurons implausible. In particular, the claim that "neurons operate in the time scale of milliseconds..." (Rumelhart 1989: p.135), which is crucial to the 100 step argument, involves a considerable oversimplification of the neurological facts. For example, cortical neurons have variety of different intrinsic firing patterns and rates (see Churchland and Sejnowski 1994: p. 53). It is also the case that the rate of firing of a particular neuron will be determined, in part, by the kind of nerve fiber which makes connections with it. deGroot and Chusid (1988: pp. 23-24) describe three distinct types of nerve fiber which have differential rates of signal conductance. Given these complexities, The simple temporal claim which is central to the 100 step argument, lacks plausibility without being defended in detail. Once again though, such a defense has not been attempted within the connectionist literature.

Even if these difficulties with the 100 step argument are overlooked, the argument still fails to unambiguously establish the conclusion its connectionist proponents propose. As Sterelny (1990: p. 172) notes, there are two possible conclusions from the 100 step argument. The weaker conclusion is that, however human brains actually work they do not run the same programs as computers do. Of course, this conclusion is almost certainly (though somewhat trivially) correct. The stronger conclusion is that the 100 step argument shows that a certain class of theories about cognition (i.e. those which are based upon GOFAI models) are fundamentally incorrect. The stronger conclusion is presumably the one which connectionists wish to endorse. The stronger conclusion is highly problematic, however.

The strong conclusion of the 100 step argument should persuade us that explanations of cognitive phenomena which are rooted in serial processing are defective. However, this alone is not sufficient to justify the adoption of a connectionist approach, rather than a GOFAI one. There are examples of GOFAI models which are parallel. Sterelny (1990: p. 172) mentions (although he does not give a reference) that "...some version of the 'Marcus parser', which models sentence comprehension by incorporating a Chomskian transformational grammar, use parallel processes." A similar point is made in both

Pylyshyn (1984) and Fodor and Pylyshyn (1988). It is also the case that the argument could turn out to be just as fatal to the plausibility of connectionist systems, if it could be shown that they required *too few* basic operations to compute certain functions. Connectionist advocates of the 100 step argument do not discuss this possibility though.

Another difficulty with the strong conclusion is that it is far from clear at what level the excessive (i.e. those in excess of 100) number of steps is supposed to arise. Would a computer program which involved more than 100 function calls be deemed unacceptable? Are the '100 steps' supposed to be basic processor operations? Without a clearer notion of what is to count as a step, it is hard to tell how the 100 step constraint could even be met by a serial processing system!

Given the problems just raised, it is reasonable to conclude that the 100 step argument should not be taken as providing support for the claim that connectionist systems are consistent with real time constraints upon processing. This, at least as a general claim about connectionist systems, is just another connectionist myth.

Myth 3: 'Connectionist Systems Exhibit Graceful Degradation'

The connectionist myths discussed above have focused primarily upon the supposed similarities between connectionist systems and biological entities. There is however another species of myths which concentrate upon attempting to show that connectionist systems are in some way preferable to GOFAI ones. These two types of myth are not totally distinct though. The claim about real time constraints discussed above for example, involves elements of both kinds of myth. I shall now consider a few of the connectionist myths which are supposed to foster support for the belief that connectionist systems are to be preferred to their GOFAI counterparts.

A cognitive system which has to interact with the real world is often faced with imperfect input data. For example, humans by and large are pretty good at reading one another's handwriting, even though handwriting usually looks very different from the block print which most people initially learn to read. Similarly, we are also pretty good at understanding what is being said to us by someone even if the speaker has a heavy accent, or the context of utterance is such that part of the utterance is obscured by background noise. We still succeed in identifying everyday objects even when they are viewed under unusual lighting conditions, or when they are viewed from unfamiliar angles. This being the case, a desirable property of computational models of cognitive processes is that such models should also be able to deal with degenerate input. Ideally, when a system is faced with incomplete, corrupt or even inconsistent input, the system should be able to make intelligent guesses about what the input should have been and make appropriate responses accordingly. If one briefly glimpses out of the corner of ones eye a bear charging towards one, waiting for more information is not an especially helpful response! The ability to handle incomplete, inconsistent or otherwise imperfect input data is sometimes called 'graceful degradation' (See Clark 1991: p. 62).[5]

One advantage often claimed for connectionist systems over their traditional CTM based counterparts is that connectionist systems exhibit graceful degradation, whilst GOFAI systems do not. Churchland (1990: p. 120) makes the point thus,

---

[5] Actually, the notion of 'graceful degradation' is somewhat more technical than this. Clark's (1991) characterization will suffice for current purposes though.

> ...you can recognize a photo of your best friend's face, in any of a wide range of poses, in less than half a second. But such a recognitional achievement still eludes the best [GOFAI] pattern-recognition programs available,...

Churchland (1990: p. 120) goes on to note that "even strongly simplified recognitional problems" are very difficult indeed for GOFAI systems. Similar claims can be found in McClelland, Rumelhart and Hinton (1987).

Now the mythological component here derives not so much from the facts, so much as the conclusion which is drawn from these facts. From the fact that many GOFAI systems, at the present time, do not exhibit graceful degradation, it does not follow that they cannot *in principle* be made to exhibit this property. The facts amount to nothing more persuasive than *prima facie* evidence. They certainly cannot be used to support the more general conclusion that connectionist systems are preferable to GOFAI ones. There are a (at least) two reasons why this is the case. First, GOFAI systems which exhibit graceful degradation to the same degree that connectionist systems apparently do, may be developed at any time.[6] Indeed, this is an area of active research. For example, systems known as 'Truth Maintenance Systems' (See Forbus and de Kleer 1992) have been developed which are able to reason effectively on the basis of incomplete information. Second, just because one type of system is apparently superior to another type with respect to one set of properties, does not mean that such a system is superior with respect to *all* relevant properties. It may well be the case that there are difficulties with connectionist systems which GOFAI systems can easily over come (for example, Fodor and Pylyshyn (1988) claim that connectionist systems have significant limitations when it comes to dealing with compositionality or systematicity).

---

[6] This objection is also raised by Sterelny (1990: pp. 173-175).

Connectionists who argue in favor of their approach on the basis of the graceful degradation of their systems over look these considerations. The upshot of this is that the putative superiority of connectionist systems over GOFAI systems is not established by the simple appeal to one or two apparent properties of these systems. In order for such arguments to be persuasive, it would be necessary to consider *all* the relevant properties. Consequently, although the factual claims about the graceful degradation of connectionist systems may, at the current time, suggest that such systems may have advantages over GOFAI systems for certain types of tasks, graceful degradation alone is not sufficient to support the conclusion that connectionist systems are superior to GOFAI ones in general. It follows from this that claimed superiority of connectionist systems, based solely upon an appeal to graceful degradation, is nothing more than a myth. Of course, if 'graceful degradation' is treated as a comparative notion and it is argued that it is easier for connectionist systems to exhibit it than GOFAI ones, then the objectionable nature of such claims is considerably reduced. The older, more absolute claims have recently been replaced by the comparative claims (see for example Bates 1996 discussion of Ling 1996).

There are a variety of other allegedly desirable properties which connectionist systems are claimed to have, which GOFAI systems do not. These include being resistant to damage, being good pattern recognizers, being good at retrieving information on the basis of the content of the information and being able to handle multiple constraint satisfaction problems (See McClelland, Rumelhart and Hinton 1989, for example). All these claims however, fail to adequately support the conclusion that connectionist systems are intrinsically superior to GOFAI systems, for reasons very similar to those described above for graceful degradation. For this reason, I will not go through the arguments here. The important point I wish to urge though is that general claims about the superiority of connectionist systems over GOFAI ones, which are made on the basis of connectionist systems apparently having some desirable property which GOFAI systems apparently lack, are (generally speaking) not adequately supported. This being the case, such claims may constitute nothing more than connectionist myths.

Myth 4: 'Connectionist Systems Are Good Generalizers'

The claim that connectionist networks exhibit graceful degradation is sometimes made in conjunction with a claim that networks are good at 'generalization' (See for example, McClelland, Rumelhart and Hinton 1986; pp. 29-30). As is the case with the graceful degradation claim, a commonly implied conclusion from the generalization claim is that connectionist systems are to be preferred to GOFAI systems as models of cognitive function. This claim is a little more interesting than the graceful degradation claim (and related claims), as such it deserves special treatment (this is not to say that the objections outlined above may also apply to this claim). However, like the connectionist claims above, the generalization claim has a mythological component.

As a rough first approximation, a system can be said to generalize when it can produce outputs which are appropriate for a particular input or class of inputs, which it has not been previously given information about. The first difficulty with the claim about the generalization of connectionist systems is that it is often the case that generalization is specified in a manner which is only appropriate for connectionist systems (or some subset of connectionist systems). For example, Clark (1993: p. 21) describes generalization thus,

> A net is said to generalize if it can treat *novel* cases sensibly, courtesy of its past training.

This notion of generalization is inordinately narrow though. For example, it would not be applicable to systems which do not undergo training. The famous Jets and Sharks network (described in McClelland, Rumelhart and Hinton 1987: pp. 26-31, McClelland and Rumelhart 1988: pp. 38-46, Clark 1991: pp. 86-92, and Bechtel and Abrahamsen 1991: pp. 21-34) is said to exhibit 'generalization' (albeit, not very good generalization, in this case), yet does not undergo training.[7] If generalization is specified broadly though (for example, as I do with the 'rough, first approximation' above), then many GOFAI systems would seem to exhibit generalization too. For example, Rip's (1983) ANDS system, which is a paradigm example of a GOFAI system, might plausibly be said to generalize in this sense.[8] This being the case, an appeal to generalization cannot adequately support the contention that connectionist systems are preferable to GOFAI ones.

It is also the case that, even if a narrow conception of generalization such as Clark's is employed, only some connectionist networks exhibit this property. A common difficulty encountered by network researchers is that, even with identical network architectures, training regimes and similar starting parameters, different versions of the same network will exhibit different degrees of generalization, due to the practice of setting initial weight and biases values randomly. Sometimes, if a network has too many hidden units and is trained to too strict a convergence criterion, a network may simply instantiate a 'look-up table' for the training set, and produce responses upon generalization testing which are equal to, or worse than mere chance! Generalization (no matter which conception is employed) is a property of only some networks, and not a general property of *all* networks.

---

[7] In fact, Clark (1991: p. 92) even makes a claim about the generalization abilities of the Jets and Sharks network.

[8] Actually, the situation is somewhat more complex than this, in so much as it is unclear whether or not the inference rules within ANDS are to count as containing information about *every* inference of a particular syntactic type. This complication does not effect my main point though, so I will not discuss it further here.

An additional complication which arises with the claim about generalization is that it is very sensitive to the particular task being considered. This fact is frequently not made explicit in the descriptions of generalization in the connectionist literature though. This is very nicely exemplified by another example from Clark. Clark (1993: p. 21) briefly describes a connectionist network, originally due to McClelland and Rumelhart (1987, Vol. 2: pp. 170-215), which was trained to recognize dogs and was trained upon sets of dog features which were supposed to correspond to the features of individual dogs. Clark (1993: p. 21) cites as an example of generalization (in accordance with the conception quoted above) the fact that,

> ...a novel instance of a dog (say, one with three legs) will still be expected to bark so long as it shares enough of the doggy central tendencies to activate the knowledge about prototypical dogs.

Although the facts are correct, this example does not do justice to the influence of the chosen task domain upon generalization. Suppose that the network was trained not only to recognize dogs, but was also trained to recognize common items of furniture. Now, if the only three legged object in the entire training set were to be a small stool, it is quite possible that the network would classify the three legged dog as a non-barking object (i.e. like a stool), rather than as a barking one. The performance of such a network would be dependent upon the ratio of dogs to furniture in the entire training set, as well as various other specific details of the training regime.[9]

An additional difficulty which undercuts connectionist claims about generalization comes from a recent paper by Clark and Thornton (1996). In this paper Clark and Thornton argue that there is a whole class of problems (so called 'Type-2' problems - see Clark and Thornton 1996, for details) which connectionist networks will not, in principle, be able to exhibit any generalization whatsoever upon.

Given all the difficulties I have outlined above, it is not unreasonable to conclude that the claim that connectionist systems are good generalizers is so problematic, that in many instances, it may amount to nothing but a myth. It is far from clear that there is even a uniform notion of generalization which is used amongst connectionists, let alone a conception which is common to both connectionism and GOFAI.[10] Without detailed clarification of the notion, it is not a suitable basis for comparison at all. Moreover, the actual evidence for the claim that networks are good generalisers is far from unequivocal. Thus, claims about generalization cannot provide an adequate basis to judge between connectionist and GOFAI systems. Furthermore, although some connectionist systems may exhibit something which might plausibly be termed generalization, it is certainly not universally the case. Thus, claims about connectionist systems and generalization which are not very carefully hedged (as they almost never are) are going to be false of many connectionist systems. Hence, the unqualified claim that connectionist systems are good generalisers is largely a myth.

---

[9] Cf. the example of a network for assessing bank loan applications, discussed by Clark (1993: p. 71).

[10] Indeed, the considerations discussed above might be taken as being indicative that the term 'generalization' exhibits what Waismann (1951) calls 'open texture'.

Not all connectionist myths, concerning the desirable properties which networks are supposed to have, are as vague as the claims about generalization. Historically, networks have been subject to detailed technical criticism, with respect to their computational power. Perhaps the best known criticism was that offered by Minsky and Papert (1969) in their book Perceptrons.[11]

Myth 5: 'Recent Connectionists Systems Have Shown That Minsky and Papert Were Wrong'

In their book Perceptrons (1969), Minsky and Papert argued that Rosenblatt's perceptrons (an early kind of connectionist system) were subject to a number of significant limitations. One of the claims of the recent connectionist researchers is that these limitations have been overcome. For example, Rumelhart, Hinton and McClelland (1987, Vol. 1: pp. 65-66) claim that,

> It was the limitations on what perceptrons could possibly learn that led to Minsky and Papert's (1969) pessimistic evaluation of the perceptron....As we shall see in the course of this book, the limitations of the one-step perceptron in no way apply to the more complex networks.

---

[11] Minsky and Papert's role in the history of network research is significant. It will be discussed in some detail in a later chapter.

As with the other myths discussed above, although there is a grain of truth which lies at the heart of this myth, strictly speaking, the above claim is largely false. Despite the facts though, the claim that modern connectionist systems have shown that Minsky and Papert were incorrect in their assessment of networks has become widely accepted. Clark (1989: p. 85) for example, describes the contemporary enthusiasm for,

> ...the work of a recent wave of connectionists who found ways to overcome many of the problems and limitations of the linear-thresholded architectures of perceptrons.

This particular myth seems to have its origins in a none too careful reading of Minsky and Papert's conclusions. Bechtel and Abrahamsen (1991: p. 15) describe a pretty typical (C.f. Rumelhart, Hinton and Williams 1987) reading of their conclusions,

> The centerpiece of their [Minsky and Papert's] criticism was their demonstration that there are certain functions,..., which cannot be evaluated by such a network [i.e. a two-layer perceptron]. An example is the logical operation of *exclusive or* (XOR). While Minsky and Papert recognized that XOR could be computed by...a multi-layered network, they raised an additional problem: there were no training procedures for multi-layered networks that could be shown to converge on a solution.

Now, if this really was Minsky and Papert's conclusion, then the demonstration of a training procedure which could converge on a solution for the XOR problem would seem to suffice to show that their conclusions were indeed wrong. Furthermore, Rumelhart, Hinton and Williams (1987) describe just such a result. Unfortunately, this is *not* an accurate description of Minsky and Papert's conclusion.

Minsky and Papert (1969: pp.3-5 and pp.22-30) are careful to specify the class of devices which they intend to study in Perceptrons. The scope of their conclusions were specifically limited to networks of linear threshold units without any feedback loops. The units standardly employed in modern connectionist networks are not linear threshold elements (for example, many units have continuous sigmoidal activation functions). Strictly speaking, such units are beyond the scope of Minsky and Papert's conclusions. Indeed, it was the adoption of continuously valued activation functions which enabled training procedures for multilayed networks to be derived. Given these facts, it is pretty clear that recent work in connectionism has not succeeded in showing that Minsky and Papert's conclusions were false.

Minsky and Papert in fact do raise a number of points in Perceptrons which are salient to modern work on network systems though. Perhaps the most significant of these concerns is what has come to be known as 'the limited order constraint'. Minsky and Papert's (1969: pp. 5-14 and 30-32) discussion of limited order is both complex and technical. In order to avoid these technicalities, I will offer a rough and ready version of this constraint. This will suffice for current purposes. The limited order constraint just amounts to the condition that the units in one layer of a network do not have connections to all the units in the next layer.

One reason why the limited order constraint is reasonable and should be considered significant (at least by connectionists interested in cognitive modeling) was mentioned earlier; biological brains seem to satisfy this constraint. That is to say, layers of neurons have comparatively sparse patterns of interconnection between them. Unfortunately though, standard connectionist practice involves violating this constraint. The limited order constraint is important to many of Minsky and Papert's conclusions. In fact, even the networks discussed by Minsky and Papert (1969: p. 250) can evaluate XOR and connectedness, if this constraint is violated. This being the case, it is no surprise that networks of modern connectionist units, with their more complex activation functions which also violate the limited order constraint can be used to construct networks which can evaluate functions which Minsky and Papert show to be beyond the systems they consider. Without the limited order constraint, the difficulty of solving many classes of problems is greatly reduced. The important point in the current context is that the results from modern connectionist systems do not serve to show that Minsky and Papert were wrong. The problems which the modern systems solve, although similar, are not the same as those considered by Minsky and Papert.[12] Given these facts, it should be reasonably clear that the connectionist claim that their systems serve to show that Minsky and Papert were wrong is, like the other claims discussed in this chapter, nothing more than a myth.

Conclusion

Each of the five myths which have been discussed in this chapter would offer some support for the contention that connectionist systems offer a good means of modeling cognitive function, were it to be the case that they were true. Similarly, many of the myths would also offer grounds for considering connectionist models superior to GOFAI ones, again subject to the condition that the myths were true. However, as I hope has become clear through the above discussion, each of the claims contains a substantially mythological component. It is only if these claims are very carefully qualified and selectively applied that they are truths. Under any other condition, especially if the claims are made as being putative general truths about the class of connectionist systems as a whole, the claims are just myths.

---

[12] Another important problem raised by Minsky and Papert which modern connectionists have a bad habit of overlooking what is known as 'The scaling problem'. Although networks may work well for small 'toy' problems, Minsky and Papert argue that networks will rapidly become unmanageably massive when faced with more complex problems. Connectionists seem to be happy to make generalizations on the basis of their toy systems, whilst ignoring this difficulty.

This conclusion is important, because the myths are often employed to support the further conclusion that network models offer a better means of modeling biological cognitive functioning than GOFAI models do. Notice however, that this second conclusion presupposes that there is are, in fact, two distinct classes of models. Although this fact might seem intuitively obvious, given the apparent difference between network models and the kinds of devices associated with the traditional interpretation of the CTM, it has yet to be substantiated. As should be clear by now, the myths of connectionism do not provide such a substantiation.

It seems to me that the lesson to be learned from the myths of connectionism is that generalizations about the properties of such systems have to be very carefully stated and conservatively made. This being the case, there are good grounds for proceeding by considering particular systems in detail and determining the significant properties of those systems, with respect to the traditional CTM. Although this is not as methodologically straightforward as one might wish, it *is* both a useful and feasible strategy. In the next chapter, I will discuss one particular connectionist system in detail. The purpose of this discussion is to get a clear picture of the system, such that an assessment of the extent to which the systems shares the properties associated with the traditional CTM can be made. Doing this should, in turn, throw light upon the nature of the putative proposed alternative to the CTM, which networks are supposed to give rise to.

V

An Empirical Study: The interpretation of the Logic Network, L10

Introduction

In the previous chapter, a number of 'myths' about connectionist systems were discussed and shown to be problematic. There is one further myth though which was not considered there. It concerns a series of related claims which have been made about representations (or 'tokens' in the terminology of Chapter 2) and the operations which manipulate such entities in connectionist systems. The purpose of discussing this myth separately from all the other myths is two-fold. First, this myth cannot be dismissed as summarily as the myths in the previous chapter. As will become clear, clarifying what is going on with this set of claims requires further exposition and consideration of a range of topics. As such, this myth is best discussed in isolation from the others. Second, as I think that this myth relates much more directly to the putative challenge which connectionist systems are supposed to present to the CCTM than do the other myths, it seems appropriate for me to treat this myth separately and in greater detail than the myths discussed so far.

Rules, Representations and Connectionist Systems

The claims quoted at the end of Chapter III from Clark (1989: p. 124), Sterelny (1990: p. 168) and Cummins (1989: p. 157, fn. 6) have a 'family resemblance' to one another, in so much as they deal with either representations or the operations which manipulate those representations. A reasonably representative statement of the myth can be found in Clark and Lutz (1992a: p. 12), when they remark,

> Connectionist models...differ from those of conventional AI [i.e. systems which are straightforwardly compatible with the CCTM] in (amongst other

> things) appearing to operate without traditional symbolic data structures over which computational operations may be defined.

Similar claims can be found in many places in the connectionist literature. For example, Rumelhart, Hinton and McClelland (1986: pp. 75-76), Bechtel and Abrahamsen (1991: pp. 151-163, *passim*), Fodor and Pylyshyn (1988: p. 5) Schneider (1987: p. 74) and Smolensky (1988: p. 1), all make roughly this claim.

There are two distinct components to this myth. The first concerns whether or not there are entities similar to 'traditional symbolic data structures' which can be said to play a crucial role in the functioning of connectionist systems. The central claim here is that the representational/token/symbol structures of networks are significantly different in kind from the items which play (roughly) the same role in the CCTM. The second component of the myth concerns the operations which occur within a network. Here, the central claim is that networks operations are significantly different from those supposed by the CCTM. As Clark and Lutz's remark cited above is not entirely unequivocal with respect to such operations, it is worth drawing upon another source for a univocal statement of this component of the myth. Churchland (1989: p. 170) claims of a system called 'NETtalk', that one of the most important features of the system is the fact that the network,

> ...contains no explicit representations of any *rules*, however much it might seem to be following a set of rules.

Claims pertaining to operations (or 'rules') and representations (or 'symbols') in connectionist networks lie at the heart of the current myth and are closely intertwined with one another in the literature. This being the case, provides grounds for considering both components of the myth together, rather than separately. In addition, as it turns out that there is common problem which all claims of this nature (regardless of their emphasis) must face up to, this provides further reasons for treating the myth as a single entity.[1]

The problem with the claims which constitute this myth is epistemological in nature and methodological in origin. To put it simply, current connectionist practice is such that it does not provide sufficient evidence to support this claim.[2] As a consequence, we really have no idea whether what kind of rules, representations and the like are deployed by connectionist networks. Indeed, the reluctance of connectionists to analyze the internal structure of their networks has been a central concern of some recent critics (e.g. McCloskey, 1991) who argue that connectionism may not be able to contribute to cognitive science. Such criticisms are considered in detail in the next section.

McCloskey's Critique of the Connectionist Research Program

McCloskey (1991) considers, in detail, the relationship between recent connectionist models and cognitive theorizing. His conclusions are, by and large, pessimistic. McCloskey (1991: p. 387) concludes that,

---

[1] It is also the case that certain related, though distinct, claims about the alleged 'autonomy' of networks also focus upon similar issues. I will not pursue these particular claims further here however. See Dawson and Schopflocher (1992) for further details.

[2] It might be thought that there is a difference in principle between connectionist representations and those of the CCTM because, in theory, any individual connectionist processing unit can assume an infinite number of distinct states (activation levels), each of which could be construed as being representational, whereas the CCTM is committed to there being only a finite number of such states (see Smolensky 1994). However, as any *actual* connectionist system will always contain a finite number of units, each one of which takes on some specific value, it follows that there is only a finite number of distinct states that any such systems can have. And so the issue then becomes one of the nature of these finite states. This is an issue which requires empirical investigation.

> ...connectionist networks should not be viewed as theories of human cognitive functions, or as simulations of theories, or even as demonstrations of specific theoretical points.

Throughout his discussion, McCloskey focuses upon one particular connectionist model, the word recognition and naming network described by Seidenberg and McClelland (1989). However, McCloskey (1991: p. 387) takes his conclusions to apply to the class of all networks which employ distributed representations, include hidden units and have connection weights set by a training procedure.

McCloskey argues that networks such as Seidenberg and McClelland's do not qualify as cognitive theories themselves. He argues this point by analogy with a black box. McCloskey supposes that there is a black box which appears to model a cognitive phenomenon pretty well. That is to say the black box's performance on the task roughly mimics (though does not quite duplicate) the performance of human subjects on the same task. Fairly obviously, unless there was a description (in terms of the structure and functioning of the components of the box) of exactly *how* such a device produced outputs as a result of particular inputs, the box would be nothing more than an interesting artifact. The box would certainly not constitute a theory. According to McCloskey, network models are frequently treated in a manner similar to such a black box. Although they can perform interesting tasks, they do not support an explanation of *how* they perform these tasks. As such, networks themselves do not constitute theories.

McCloskey also maintains that even if additional information, such as a detailed description of the network architecture, the input and output representations employed, information on the functioning of individual units and the details of the training procedure employed are supplied, the network still does not amount to a theory. For, he says (1991: p. 388), as there are still questions which the network cannot be used to answer, the network itself does not constitute a theory. Such questions as the relevance of the representations employed to the performance of the network need to be answerable before the network can count as a theory, in McCloskey's opinion.

According to McCloskey (1991: p. 389), connectionism fails to provide information useful to cognitive theorizing for two reasons. First, connectionist 'theories' are not stated in sufficient detail. Second, there are serious problems with regards to tying particular theoretical proposals to implemented networks. To illustrate these points, McCloskey notes that Seidenberg and McClelland's network and so-called theory fails to explain exactly how their network deploys the appropriate knowledge under just the appropriate circumstances. There is nothing in the theory which accompanies Seidenberg and McClelland's network, for example, which can account for how the letter 'a' should be processed so as to distinguish its use between "gave" and "have".

Of course, the way that connectionists could meet McCloskey's challenges would be to claim that the network itself is supposed to provide the details of the theory. Although initially appealing, such a simple response is not satisfactory. McCloskey (1991: p. 390) notes that any simulation involves the implementation of both theory-relevant and theory-irrelevant details, and as a consequence there is no way to determine which aspect of a model are significant for the theory and which are not. For example, the ‘theory’ (so construed) makes no distinction between issues such as the learning rate employed in a simulation (which is unlikely to be too theoretically significant) and the pattern of weights of the trained network (which is likely to be highly theoretically significant), rather it simply implements them both. If, say, a network implemented a theory of how to solve some logic problems, the network would have both employed a learning rate and generated a final pattern of weights. Yet the pattern of weights would be crucial to which theory was implemented, whereas the learning rate would be irrelevant.

McCloskey's point here seems to just amount to an appeal to the familiar observation concerning the underdetermination of theory by evidence; even by all *possible* evidence (See Quine 1951, Glymour 1980 and Kitcher 1993: pp. 247-249 for a detailed discussion). Although this might initially appear to blunt the force of McCloskey's criticism, it should not be taken as doing so. In many cases of underdetermination, sensible choices can be made with respect to which theories to accept and which to reject on the basis of considerations such as admissible cost functions, prior practice and the like (see Kithcher 1993: pp. 250 - 252). However, in the case of attempts to determine precisely which cognitive theory a particular network simulation instatiates no such additional constraints appear to be straightforwardly applicable. For example, how could one apply a cost function to a claim that a particular network instantiates one theory rather than another, especially if the two theories are similar to one another?

McCloskey also raises another problem which is of deeper significance. McCloskey (1991: p. 390) describes the difficulty thus;

> ...the problem is that connectionist networks of any significant size are complex nonlinear systems, the dynamics of which are extremely difficult to analyze and apprehend.... At present, understanding of these systems is simply inadequate to support a detailed description of a network's knowledge and functioning.

Indeed, this observation is perhaps the most important point of McCloskey's critique of connectionism. Although there are techniques for analyzing trained networks, they are, in McCloskey's evaluation, not sensitive enough to provide the detailed information that would be required of a serious candidate for a theory. This fact also makes deciding whether or not a particular network actually implements a particular cognitive theory impossible to determine. And so, the justification of claims about the relationship between networks and the CCTM are impossible to clarify in any detail.

If McCloskey is correct then, it would seem that, despite the potential contribution of connectionist networks to cognitive theory, in practice such networks cannot make any such contribution. On the one hand, if networks are taken as theories themselves, then there is no way of determining exactly what the details of this theory are. On the other hand, if a network is supposed to be an implementation of a particular theory, then there is no way of determining whether or not the network actually succeeds in implementing that theory. In both cases, the difficulties derive from the fact that there is no way of understanding the details of the networks knowledge and functioning. Like black boxes, the networks cannot be used to elucidate the structures and functions needed to explain how the task is performed. As a consequence, even if it were to be the case that networks provided a basis upon which an alternative to the CCTM could be developed, the precise nature and details of that alternative would be radically unclear.

Given the difficulties surrounding evidence from connectionist systems informing cognitive theory, it also seems reasonable to wonder about the status of the theoretical claims advanced by connectionists on the basis of their models. Bechtel and Abrahamsen (1990), for example, describe two networks which they trained upon logic problems. The networks performed the tasks moderately well once they were trained, though not perfectly. On the basis of the networks performance, Bechtel and Abrahamsen (1990: p. 173) conclude that,

> The ability to reason using logical principles may not need to be grounded in proposition-like rules...

If this claim were true, then it would suggest that networks differ from the CCTM, at least with respect to property (3) of the CCTM, described in Chapter II.[3] Since Bechtel and Abrahamsen offer no analysis of the structure of their trained networks, the basis for this conclusion seems somewhat mysterious. After all, if they have no idea about *how* their networks go about solving the problems, it seems odd to make such a claim. The networks *could* be employing proposition like rules, but without analysis there is no way of knowing one way or the other. Bechtel and Abrahamsen's conclusion consequently appears to be unwarranted on the basis of the evidence they present. Their conclusion appears only to follow if one subscribes to the myth about the representational structures and rules in connectionist systems.

So, it seems that McCloskey's critique has two consequences. First, it indicates that even if connectionist systems do provide the basis of a genuine alternative to the CCTM, contemporary connectionist methodology can give little evidence about the nature of this alternative. Second, it seems that the possibility of connectionist models being able to make a contribution to cognitive theorizing is threatened by the fact that networks are treated as black boxes. These consequences suggest that understanding trained connectionist systems is of crucial importance to connectionists, and so we turn to that topic.

---

[3] Property (3) is the property of having "A capacity to perform a determinate range of precise and exceptionless operations upon tokens".

Understanding Trained Connectionist Networks

McCloskey is unimpressed by current attempts at the analysis of trained connectionist networks. His (1991: p. 309) assessment is that "...techniques for network analysis are currently rather crude." Furthermore, he is not optimistic about the future prospects of analytic techniques. In McCloskey's (1991: p. 394) opinion,

> ...it is not clear how fast and how far we will progress in attempting to analyze connectionist networks at levels relevant for cognitive theorizing.

Robinson (1992: p. 655), urges a similar conclusion and suggests that,

> We may have to accept the inexplicable nature of mature networks.

When Hecht-Nielson (1990: p. 10) considers the future prospects of analyzing networks and trying to answer questions about exactly how networks produce the results they do, he notes that,

> ...there is a growing suspicion that discovering answers to questions of this type may require an intellectual revolution in information processing as profound as that in physics brought about by the Copenhagen interpretation of quantum mechanics.

Mozer and Smolensky (1989: p. 3) make the same point more colorfully when they note that,

> ...one thing that connectionist networks have in common with brains is that if you open them up and peer inside, all you can see is a big pile of goo.

If this widespread pessimism about the possibility of analyzing and interpreting trained networks is well founded, then the prospects of networks being able to play theoretically significant role in cognitive science, so as to provide the basis of a challenge to the CCTM would seem to be similarly adversely affected.

However, the pessimism expressed above notwithstanding, there are a number of techniques which have been developed and applied to the analysis and interpretation of trained connectionist systems. Many different types of techniques exist for this type of analysis. For example, Hanson and Burr (1990) review a number of techniques for analyzing weights, including compiling frequency distributions of connection strengths, quantifying global patterns of connectivity with 'star diagrams', and performing cluster analyses of hidden unit activations. Techniques frequently employ statistical approaches and factor analytic strategies. Indeed, these approaches have even received some discussion in the philosophical literature (See Clark 1993).

Another analytic technique was recently described by Berkeley, Dawson, Medler, Schopflocher and Hornsby (1995). This approach seems to offer considerable promise in revealing the kinds of information about trained networks which would be salient to the relation between connectionist systems and the CCTM, and so I will consider and describe it, and some of the results obtained using it, in detail.

In order to illustrate this analytic technique in practice, it is worth considering its application to a particular network. To this end, I will discuss the network known as 'L10', that Berkeley *et al.* trained to solve a set of logic problems, originally studied by Bechtel and Abrahamsen (1991). I shall begin by describing the problem set in a little detail.

Bechtel and Abrahamsen's Logic Problem

Bechtel and Abrahamsen (1991) and Bechtel (1994) describe a logic problem which they trained a network to solve. All the input patterns had two premises and a conclusion. The first premise contained two variables, both of which could be negated, and a connective. All the second premises and the conclusions were made up of single variable letters, which could also be negated. There were three possible connectives in the first premise, IF...THEN..., ...OR... and NOT BOTH...AND.... This meant that there were four distinct classes of problems (Modus Ponens, Modus Tollens, Alternative Syllogism and Disjunctive Syllogism), although there were two versions of both the Alternative Syllogism and the Disjunctive Syllogism types.

Different examples of each argument type were constructed with four possible values (A,B,C,D) for the variables in the premises and conclusion. As the variables could be negated, this gave rise to 48 valid and 48 invalid instances for the Modus Ponens and Modus Tollens problem types and both kinds of the two Alternative Syllogism and the Disjunctive Syllogism types. Thus in total, the training set consisted of 576 patterns (i.e. (48 + 48) * 6). The task for the network, after training, was to be able to identify the type of problem and determine whether or not a particular argument was valid. Table 5-1 gives an example of a valid instance of each problem type.

Problem Type	Problem Example	Descriptive Notation
Modus Ponens (MP)	If A Then B A -------------- Therefore B	Connective: If...then... S1(V1): A S1(V2): B S2: A C: B
Modus Tollens (MT)	If A Then C Not C --------------- Therefore Not A	Connective: If...Then... S1(V1): A S1(V2): C S2: C S2 is negated C: A C is negated
Alternative Syllogism (AS) Type 1	D Or A Not D --------------- Therefore A	Connective: ...Or... S1(V1): D S1(V2): A S2: D S2 is negated C: A
Alternative Syllogism (AS) Type 2	B Or C Not C --------------- Therefore B	S1(V1): B S1(V2): C S2: C S2 is negated C: B
Disjunctive Syllogism (DS) Type 1	Not Both C and D C ----------------------- Therefore Not D	S1(V1): C S1(V2): D S2: C C: D C is negated
Disjunctive Syllogism (DS) Type 2	Not Both A and D D ----------------------- Therefore Not A	S1(V1): A S1(V2): D S2: D C: A C is negated

Table 5-1
Examples of valid inferences from Bechtel and Abrahamsen's (1991) logic problem set. Notation: S1(V1) - The first argument place (V1) of the first premise (S1) of an argument. S1(V2) - The second argument place (V2) of the first premise (S1) of an argument. S2 - The second premise of an argument. C - The conclusion of an argument.

In order to facilitate easy discussion of the interpretation of the network Berkeley *et al.* developed a special descriptive notation. This too is introduced in Table 5-1. In this descriptive notation, each of the argument places in the problems was assigned a unique descriptive code. In addition to these letter codes, other important information, such as the type of connective in the first premise and whether or not particular letters are negated, can also be represented in the full description of a particular problem in the descriptive notation (as illustrated in Table 5-1). It is also necessary to be able to compare particular pairs of variables, with respect to whether they were both negated or not. Berkeley *et al.* describe these relationships in terms of the variables relative 'signs'. If two variables are both negated or both non-negated, then they are deemed to be of the 'same sign'. Otherwise, the variables were said to be of 'opposite sign'.

The Network L10

Bechtel and Abrahamsen's original network for this logic problem had two layers of hidden units, each with ten units in it. The processing units which they used were the standard kind, with sigmoidal activation functions. However, Berkeley *et al.* (1995) found that by using a network constructed from Dawson and Schopflocher's (1992) value units, the task could be learned by a network with a single layer of ten hidden units.[4] It was a network of this architecture which Berkeley *et al.* (1995) successfully trained upon the problem and which they then analyzed and interpreted. For ease of reference, I will call this network 'L10'.

---

[4] See Chapter III for a brief discussion of the differences between these kinds of units.

The L10 network had fourteen input units, three output units and ten hidden units. Each of the units in each layer had modifiable weighted connections to each of the units in the next layer. The pattern of interconnection between the processing units in the network is illustrated in Figure 5-1.

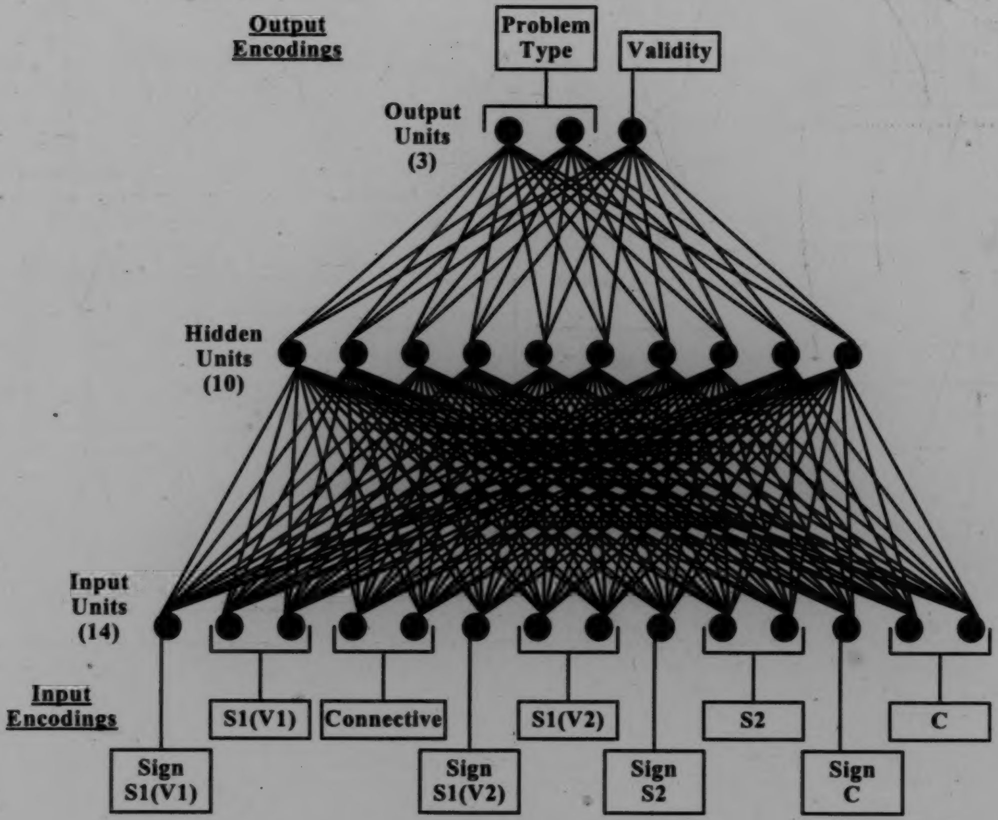
Output Encodings
Problem Type
Validity
Output Units (3)
Hidden Units (10)
Input Units (14)
Input Encodings
S1(V1)
Connective
S1(V2)
S2
C
Sign S1(V1)
Sign S1(V2)
Sign S2
Sign C

Figure 5-1
The pattern of interconnections between processing units and the input and output encoding scheme used with network L10.

The Problem Encoding Scheme of L10

Berkeley *et al.* (1995) encoded the problems of the training set upon the 14 input units, using the representational scheme devised by Bechtel and Abrahamsen (1991). This too is illustrated in Figure 5-1. The four variables in each particular problem were encoded across pairs of units. Numbering from left to right of the network diagram, input units 1 & 2, 6 & 7, 9 & 10 and 12 & 13 encoded the four possible variables. When the variable 'A' occurred in the training set in a particular argument place, the pair of input units which represented that argument place were set to '0 1'. When the variable 'B' occurred in the training set in a particular argument place, the pair of input units which represented that argument place were set to '1 0'. When the variable 'C' occurred in the training set in a particular argument place, the pair of input units which represented that argument place were set to '1 1'. Finally, when the variable 'D' occurred in the training set in a particular argument place, the pair of input units which represented that argument place were set to '0 0'. The assignment of particular bit patterns to particular variables was essentially arbitrary. The significant point was only that each letter had a unique encoding. Which particular bit pattern was assigned to which letter was of no consequence.

Whether or not particular variables were negated was indicated by single units adjacent to the unit pairs representing each variables. Thus, input units 0, 5, 8 and 11 all indicated the signs of the relevant variables. If a letter was negated, the negation units were set to 1, otherwise they were set to 0.

Finally, two units (3 and 4) were used to encode the three possible connectives in a manner similar to the way that two units encoded letters. Problems containing IF...THEN... as the main connective had units 3 and 4 set at '1 1'. Problems containing ...OR... as the main connective had units 3 and 4 set at '0 1'. Problems containing NOT BOTH...AND... as the main connective had units 3 and 4 set at '1 0'. As there were only three possible connectives in the training set, the encoding '0 0' was not used.

This encoding scheme ensured that each problem had a unique encoding, which consisted of a string of 14 binary bits. For example, the following problems

(a) If A Then B,	(b) Not Both Not D And A
A	Not A
-----------	----------------
B	Not D

would be represented to the network on the input layer as,

(a') 0 0 1 1 1 0 1 0 0 0 1 0 1 0
(b') 1 0 0 1 0 0 0 1 1 0 1 1 0 0

A similar set of representational conventions were employed for the three output units of the network. Two of the output units, units 0 and 1, were used to encode the problem type. The now familiar two bit encoding systems was used again. Modus Ponens problems were signified, under ideal conditions, by these two units being set to the values '0 1', respectively. Modus Tollens problems were signified, under ideal conditions, by these two units being set to the values '1 0', respectively. Alternative Syllogism problems were signified, under ideal conditions, by these two units being set to the values '1 1', respectively. Finally, Disjunctive Syllogism problems were signified, under ideal conditions, by these two units being set to the values '0 0', respectively.

Validity was represented by a single output unit. It was assigned the value of '1' to indicate a valid argument and '0' to represent an invalid argument. Further details on this encoding scheme and the training set can be found in Bechtel and Abrahamsen (1991: pp. 167-171).

The Training of L10

Berkeley *et al.* (1995) trained the L10 network using Dawson and Schopflocher's (1992) extension of the generalized delta rule. They used a learning rate of 0.03 and a momentum of 0.0. They also randomly set the connection weights and biases in the range from -0.3 to 0.3. During training, the strengths of connection weights were altered, but the biases were held constant at their randomly set values. Connection weights were changed after each pattern was presented to the network and the order in which the patterns were presented to the network was randomized after each complete presentation of the entire training set.

Berkeley *et al.* (1995) trained the L10 network until the network produced the correct response on all three output units, for every single pattern in the 576 pattern training set. They operationalised 'correct responses' such that, if the desired response of a particular output unit to a particular pattern was 1, then an activation of 0.9 or greater would count as being correct, and if the desired response was over an activation of 0.1 or less would count as being correct. The L10 network reached convergence, that is to say produced the correct response on each of the output units for every pattern in the training set, after 5793 presentations of the training set ('epochs').

The Network Analysis Technique

One standard technique for understanding aspects of brain function employed by neuroscientists is what is known as 'single unit recording' (see Churchland & Sejnowski, 1992: pp. 440-442). This technique involves the insertion of a micro-electrodes into a brain, so as to enable recordings of intracellular and extracellular potentials to be made, whilst the organism is exposed to various stimuli. Using this technique for example, neurons which are responsive to lines of various orientations in the visual field have been identified within the visual cortex of cats and monkeys (Kolb & Wishaw 1990: p. 48).

The reason for mentioning this fact here is that the analytic technique described by Berkeley *et al.* (1995) starts by requiring that a roughly analogous procedure be performed upon the hidden units of a network after it has learned to solve a particular problem. Once L10 had been trained to convergence, Berkeley *et al.* (1995) re-presented the training set to it and recorded the levels of activation of each hidden unit, for each problem in the training set.

The step of recording hidden unit responses to the training set is crucial to analytic technique described by Berkeley *et al.* (1995). The information recorded from each hidden unit was then illustrated by Berkeley *et al.* (1995) using what is known as a 'jittered density plot' (see Chambers, Cleveland, Kleiner and Tukey, 1983). It is worth pausing briefly to explain how such plots should be read.

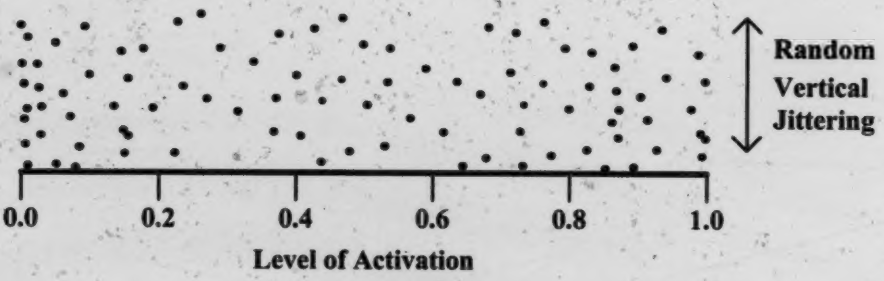

Figure 5-2
An example of a jittered density plot.

A jittered density plot consists of a number of points plotted against a horizontal axis. Each point on a particular jittered density plot, as these plots are used by Berkeley *et al.* (1995) in their analytic technique, corresponds to the level of activation in a particular hidden unit, for one input pattern in the training set. The horizontal location of each point is indicative of the level of activation which that particular input pattern caused in the particular hidden unit. Consider a particular point in such a plot. The precise horizontal location of a point on a jittered density plot is dependent upon the level of hidden unit activation associated with that point. The vertical position of a particular point, by contrast, has no such significance. When jittered density plots are generated, a random component -- the vertical 'jitter' -- is added, so as to prevent points from overlapping with one another. Thus, the height of a particular point above the horizontal axis is of no significance.

Jittered density plots are of interest in the current context, because Berkeley *et al.* (1995) showed that trained networks of value units often exhibit a marked 'banding' effect in such plots, when the information from representing the training set is displayed in this manner. This banding is crucial to technique for analyzing networks described by Berkeley *et al.* (1995).

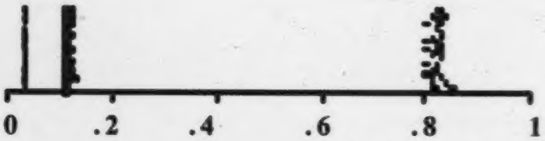

Figure 5-3
An example jittered density plot for a hidden unit of a value unit network.

Once bands have been identified, the next step in the analytic process is to find what Berkeley *et al.* (1995) call the 'definite features' associated with each band. The purpose of doing this is to identify common attributes or properties, in terms of input features, of the patterns which fall into particular bands. Definite features came in two varieties, unary and binary.

Berkeley *et al.* (1995: pp. 172-173) defined a unary definite feature as follows;

> A definite unary feature [for a particular band] was defined as an input bit that had a constant value for all the patterns within the band.

One advantage of this notion of a unary definite feature is that it permits their easy identification by use of descriptive statistics on the input patterns which fall into a particular band. A particular band has a unary definite feature just in case the mean value of a particular input for all the patterns in the band is either 1 or 0 and the standard deviation for that input is also 0. Such statistical results indicated that the particular input has zero variability and is consequently a constant for all the patterns which fell into the band.

Berkeley *et al.* (1995: p. 173) defined a binary definite feature as follows;

> A definite binary feature was defined as a perfect negative or perfect positive correlation between pairs of binary [input] features, the former representing the fact that two bits were always opposite in value, the latter representing the fact that two bits were always equal in value.

As was the case with unary definite features, this definition of binary definite features enabled Berkeley *et al.* (1995) to easily identify binary definite features by performing descriptive statistics on the input patterns which fall into a particular band. A particular band had a binary definite feature just in case there was a correlation of exactly 1 or -1 between pairs of bits in the input pattern. A binary definite feature reflects a particular relationship between the values given to pairs of input units. It also reflects the fact that this relationship holds for *all* the input patterns which fall into a particular band. For binary input data, if a correlation of 1 is found between two input bits, it indicates that the input bits have the same value for all the input patterns in the band. Similarly, if a correlation of -1 is found between two input bits, it indicates that the input bits have the opposite value for all the input patterns in the band. That is to say, when one bit has the value 1 the other will have the value 0, or when one has the value 0 the other will have the value 1.

The Analysis of Network L10

In order to illustrate this analytic technique, let us turn to the example of this technique applied to the network L10, described by Berkeley *et al.* (1995). Figure 5-4 illustrates the jittered density plots for the network's ten hidden units.

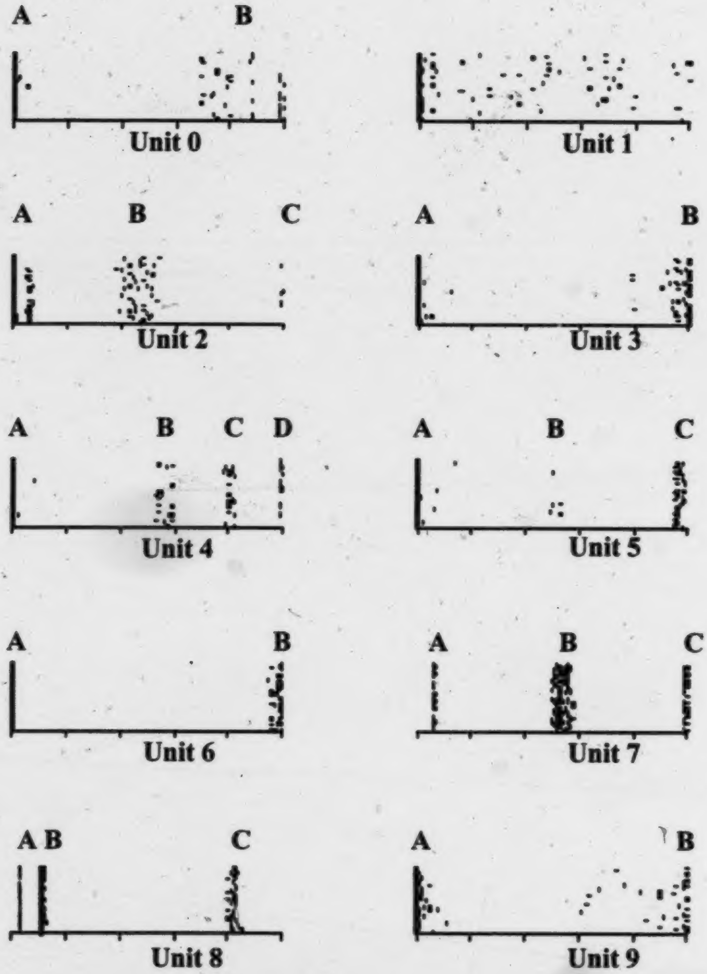

Figure 5-4
Jittered density plots for the 10 hidden units of the network L10, displaying the level of activation in each unit for each of the 576 patterns in the L10 training set.

It should be immediately apparent that the banding phenomenon is fairly clearly exhibited by all the hidden units, with the exception of unit 1. Note also the convention for naming bands which Berkeley *et al.* (1995) adopted. Individual bands in the jittered density plot for a particular hidden unit are assigned a letter, starting with the leftmost band. This facilitates easy reference to particular bands for the purposes of discussion.

In order to illustrate the procedure for identifying definite features, it is perhaps helpful to consider how such an analysis is done on one particular band, which has both unary and binary definite features. For this purpose, I will consider band B of hidden unit 4 of the network L10, described by Berkeley *et al.* (1995).

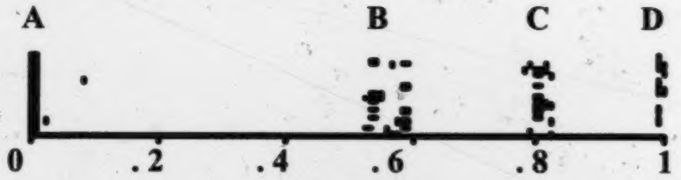

Figure 5-5
The jittered density plot for hidden unit 4 of the L10 Network.

Band B of hidden unit 4 of L10 contained just 48 out of the possible 576 patterns in the training set. When the appropriate descriptive statistics were computed with respect to the input patterns for the 48 patterns in band B, a number of definite features were revealed. Band B exhibited two unary definite features which are detailed in Table 5-2.

It turned out that the only unary definite features found for the 48 patterns in band B of hidden unit 4 of L10 were associated with input units 3 and 4. Given the fact that each input unit was assigned a particular representational role, with respect to the training set, this enabled Berkeley *et al.* (1995) to determine exactly which input features all the patterns shared in common.

Input Unit Number	**Mean Value**	**Standard Deviation**	**Interpretation**
3	1	0	IF...THEN... is the main connective
4	1	0	

Table 5-2
The unary definite features found in band B of hidden unit 4 of network L10.

In the case of the unary definite features associated with this band, the features arise for input units 3 and 4 only. As input units 3 and 4 were the two units which encoded the connective of the problem, this enabled Berkeley *et al.* (1995) to determine that all the 48 patterns which fell within this band had in common the property of having 'IF...THEN...' as the main connective.

Band B of hidden unit 4 also exhibited six binary definite features. As in the case of the unary definite features, because the input units of L10 had particular assigned representational roles, this enabled Berkeley *et al.* to determine what properties of the input that the patterns which fell into the band shared in common.

Input Unit Pair	**Correlation**	**Interpretation**
0 - 11	1	S1(V1) is the same sign as C
1 - 12	1	S1(V1) is the same letter as C
2 - 13	1	
5 - 8	1	S1(V2) is the same sign as S2
6 - 9	1	S1(V2) is the same letter as S2
7 - 10	1	

Table 5-3
The binary definite features found in band B of hidden unit 4 of network L10.

The binary definite features identified in the 48 patterns which fell into band B of hidden unit 4 of L10 are detailed in Table 5-3, along with the interpretations of those definite features. This example illustrates nicely the power of the analytic technique described by Berkeley *et al.* (1995). The process of first identifying bands, followed by identifying the definite features associated with those bands, makes it possible to come up with a reasonably detailed understanding of what particular hidden units in the network L10 are sensitive to under various input conditions, when the network solves Bechtel and Abrahamsen's (1991) logic problem. In particular, identifying definite features makes it possible to provide an interpretation of the bands by associating them with properties of input patterns.

Berkeley *et al.* (1995) reported being able to recover definite features and associated interpretations of those features for almost all the bands in the jittered density plots of the hidden units of L10, displayed in Figure 5-4. The interpretations they discovered are detailed in Table 5-4. The interpretations of the bands presented in Table 5-4 vary quite a bit with respect to their complexity. Arguably, bands with the least complex interpretations are those which just respond to the connective in the first premise, such as all the bands of hidden units 8 and 6. These bands just have unary definite features. Other bands respond to the relationships between letters in the various variable places of arguments. Band C of hidden unit 3, for example, is such that any input pattern in which the first variable of the first premise is the same letter as the variable in the conclusion and the second variable of the of the first premise is the same letter as the variable in the second premise, will cause the unit to adopt an activation which falls into this band. In this case, the band has four binary definite features which correspond to the input bits which encode the four variables.

UNIT NUMBER	BAND	NUMBER OF PATTERNS	MEDIAN ACTIVITY	INTERPRETATION OF DEFINITE FEATURES
0	A	456	0.00	No definite features
	B	72	0.77	S1(V1) is the same letter as S2 S1(V2) is the same letter as C The connective is not IF...THEN
	C	48	0.99	S1(V1) is the same letter as S2 S1(V1) is opposite in sign to S2 S1(V2) is the same letter as C S1(V2) is opposite in sign to C The connective is IF...THEN
1	A	576	0.00	No definite features
2	A	456	0.00	No definite features
	B	96	0.46	S1(V1) is the same letter as S2 S1(V1) is the same sign as S2 S1(V2) is the same letter as C The connective is not NOT BOTH...AND
	C	24	0.99	S1(V1) is the same letter as S2 S1(V1) and S2 are not negated S1(V2) is the same letter as C S1(V2) is opposite in sign to C The connective is NOT BOTH...AND
3	A	456	0.00	No definite features
	B	12	0.81	S1(V1) is negated S1(V1) is the same letter as C S2 and C are not negated S1(V2) is the same letter as S2 The connective is OR
	C	86	0.99	S1(V1) is the same letter as C S1(V2) is the same letter as S2
4	A	431	0.00	No definite features
	B	48	0.56	S1(V1) is the same letter and sign as C S1(V2) is the same letter and sign as S2 The connective is IF...THEN
	C	48	0.81	S1(V1) is the same letter and sign as C S1(V2) is the same letter as S2 S1(V2) is opposite in sign to S2 The connective is NOT BOTH...AND
	D	48	0.99	S1(V1) is the same letter and sign as C S1(V2) is the same letter as S2 S1(V2) is opposite in sign to S2 The connective is OR
5	A	456	0.00	No definite features
	B	24	0.51	S1(V1) is the same letter as C S1(V1) is opposite in sign to C S1(V2) is the same letter as S2 S1(V2) and S2 are not negated The connective is NOT BOTH...AND
	C	96	0.97	S1(V1) is the same letter as C S1(V2) is the same letter as S2 S1(V2) is opposite in sign to S2 The connective is not NOT BOTH...AND

Table 5-4a
Interpretations of the bands of the hidden units of network L10

UNIT NUMBER	BAND	NUMBER OF PATTERNS	MEDIAN ACTIVITY	INTERPRETATION OF DEFINITE FEATURES
6	A	384	0.00	The connective is not OR
	B	192	1.00	The connective is OR
7	A	96	0.06	S2 is negated The connective is NOT BOTH...AND
	B	384	0.54	The connective is not NOT BOTH...AND
	C	96	0.99	S2 is positive The connective is NOT BOTH...AND
8	A	192	0.03	The connective is OR
	B	192	0.11	The connective is IF...THEN
	C	192	0.82	The connective is NOT BOTH...AND
9	A	512	0.00	No definite features
	B	64	0.95	No definite features

Table 5-4b
Interpretations of the bands of the hidden units of network L10

Perhaps the most complex interpretations arise for bands in which units adopt activations within the band on the basis of combinations of letter similarities, relationships between the negated or unnegated status of those letters (what is referred to as sign, in the descriptive notation) and the presence or absence of particular connectives in the first premise. Band D of hidden unit 4 is an example of a band of this kind.

It is clear from Table 5-4 that there is some considerable redundancy in the set of properties which the bands are sensitive to. For example, a simple valid Modus Tollens problem with A in the antecedent position and B in the consequent position (e.g. If A Then B, Not B, Therefore Not A) would have the fact that the antecedent letter matched the letter in the conclusion and that the consequent letter matched the letter in the second premise represented (in conjunction with other information) by the pattern causing an activation in band C of unit 3 and an activation in band C of unit 5. Similar redundancy can be found for other input patterns too. Although the bands may not represent the problems in the most efficient way though, they do have the virtue (in conjunction with the network architecture) of being able to solve all the problem in the training set.

Some bands, especially A bands, which often have mean activations close to 0, are assigned the interpretations 'No definite features'. However, in most of such cases, this simply indicates that patterns that fall into the A band of a particular unit do *not* possess any of the definite features that are associated with any of the other (non-zero) bands of that unit.

Although the banding analysis technique described by Berkeley *et al.* (1995) does offer considerable insight into the way that the network L10 succeeds in solving the set of logic problems, it is not without its limitations. For example, the technique will not produce good results for problems which do not have sparse problem spaces. Another apparent limitation is exhibited by the results presented in Table 5-4. The analytic technique failed to come up with an interpretation for hidden units 1 and 9 of L10. This can be either because there were no definite features to be found in the unit's bands (as in the case of unit 9), or because the unit does not exhibit banding at all (as in the case of unit 1). However, this apparent difficulty is merely indicative that, in all likelihood, with the appropriate starting values, a network with fewer hidden units could be successfully trained upon the logic problem.[5]

Berkeley *at al.* (1995) showed that their network L10 could be analyzed in some considerable detail, using their banding analysis technique. Although the technique was not entirely perfect, the analysis of L10 goes a long way to satisfying the requirements of net analysis raised by McCloskey (1991), discussed above. As such, the analysis of the network provides a good deal of information which is germane to assessing the relationship of connectionist systems to the CCTM. However, although the definite features described above clearly indicate which input properties each hidden unit of L10 was 'paying attention' to, they do not provide much insight into the way that the bands (or their associated definite features) enable the network to solve the set of logic problems.

---

[5] Unpublished data from network simulations of this problem supports this conjecture.

Definite Features and Inference 'Rules'

Berkeley *et al.* (1995) demonstrated that it was quite feasible to characterize the problems in the training set, just in terms of the bands of hidden unit activity which they produced in the hidden layer. More interestingly though, they showed that there are useful and perhaps surprising generalizations which arise from characterizing problems with such 'band descriptions'. For example, every valid Modus Ponens problem produced a unique pattern of bands [0-A, 1-A, 2-B, 3-A, 4-A, 5-A, 6-A, 7-B, 8-B, 9-A] in the hidden layer. That is to say, every valid Modus Ponens problem produced activity of hidden unit 0 such that it fell into band A, produced activity of hidden unit 1 such that it fell into band B, and so on.

Modus Ponens was not the only type of problem which produced a unique pattern of activation, when these activations were characterized as band descriptions. Berkeley *et al.* (1995) discovered that when band descriptions were produced for the various other problem types, in a significant number of cases, there was a unique band description which every problem of the kind fell into. Given that each band is associated with a set of definite features which is in turn associated with input properties, this enabled Berkeley *et al.* (1995) to determine the set of properties the network used to solve problems of each kind. Even more significantly, once Berkeley *et al.* had discovered these sets of properties, they were then in a position to compare the network's solutions to other means of solving problems of the same kind.

PROBLEM TYPE	FORMAL DEFINITION OF RULE FOR VALID PROBLEM TYPE	NETWORK 'RULE' IDENTIFIED BY THE INTERPRETATION	NOTES ABOUT NETWORK 'RULES'
Valid Modus Ponens (MP)	S1(V1) = S2 S1(V2) = C SIGN S1(V1) = SIGN S2 SIGN S1(V2) = SIGN C CONNECTIVE: IF...THEN	S1(V1) = S2 S1(V2) = C SIGN S1(V1) = SIGN S2 CONNECTIVE: IF...THEN	The network 'rule' is the same as the formal rule except that the network does not pay attention to the signs of S1(V2) and C. Due to the nature of the training set, though, this is not necessary.
Valid Modus Tollens (MT)	S1(V1) = C S1(V2) = S2 SIGN S1(V1) ≠ SIGN C SIGN S1(V2) ≠ SIGN S2 CONNECTIVE: IF...THEN	S1(V1) = C S1(V2) = S2 SIGN S1(V2) ≠ SIGN S2 CONNECTIVE: IF...THEN	Although the network does not pay attention to the signs of S1(V1) or C, this is not significant due to the nature of the training set.
Valid Alternative Syllogism (AS) (i) [There are two versions of AS in the training set]	S1(V1) = S2 S1(V2) = C SIGN S1(V1) ≠ SIGN S2 SIGN S1(V2) = SIGN C CONNECTIVE: OR	CONNECTIVE: OR	This is a 'default rule'. Provided the connective is OR and no other definite features are true of the pattern, then the problem must be a valid AS.
(ii)	S1(V1) = C S1(V2) = S2 SIGN S1(V1) = SIGN C SIGN S1(V2) ≠ SIGN S2 CONNECTIVE: OR	S1(V1) = C S1(V2) = S2 SIGN S1(V1) = SIGN C SIGN S1(V2) ≠ SIGN S2 CONNECTIVE: OR	Here the network is sensitive to exactly the same set of properties as the traditional inference rule.
Valid Disjunctive Syllogisms (DS) (i) [There are two versions of DS in the training set]	S1(V1) = S2 S1(V2) = C SIGN S1(V1) = SIGN S2 SIGN S1(V2) ≠ SIGN S2 CONNECTIVE: NOT BOTH...AND	S2 IS NEGATED CONNECTIVE: NOT BOTH...AND	This is another 'default rule'. Provided that S2 is negated, the connective is NOT BOTH...AND, and no other definite features are present, then the problem must be a valid DS.
(ii)	S1(V1) = C S1(V2) = S2 SIGN S1(V1) ≠ SIGN C SIGN S1(V2) = SIGN S2, CONNECTIVE: NOT BOTH...AND	S1(V1) = C S1(V2) = S2 SIGN S1(V2) ≠ SIGN C SIGN S1(V2) = SIGN S2) S1(V2) IS NOT NEGATED S2 IS NOT NEGATED CONNECTIVE: NOT BOTH...AND	This network is sensitive to the same set of properties as the second traditional inference rule for DS, apart from the additional stipulation that S2 and S1(V2) are not negated.
		S1(V1) = S2 S1(V2) = C SIGN S1(V1) = SIGN S2 SIGN S1(V2) ≠ SIGN C S1(V1) IS NOT NEGATED S2 IS NOT NEGATED CONNECTIVE: NOT BOTH...AND	This network is sensitive to the same set of properties as the first traditional inference rule for DS, apart from the additional stipulation that S2 and S1(V2) are not negated.

Table 5-5

The patterns of bands produced by L10 for each problem type and the properties associated with each band, as compared to the properties associated with the inference rules of natural deduction. Grey shading indicates effective equivalence of properties. (S1(V1) and S1(V2) are the first and second variables in sentence 1; S2 is the variable in sentence 2; C is the variable in the conclusion; SIGN refers to whether a variable is negated or not negated).

Since traditional inference rules (see Bergmann *et al.* 1990) can be considered as stipulating a specific set of relationship between variables, signs and connectives, then they can be straightforwardly expressed in the descriptive notation introduced by Berkeley *et al.* (1995) and described above. Berkeley *et al.* (1995) did just this so as to provide a simple means by which L10's solution to the problem could be directly compared with the properties and relations stipulated by the traditional rules. It turned out that there are significant similarities between the set of properties and relationships to which L10 is sensitive to and which it uses to operate upon problems of the various types, and the set of properties and relationships stipulated by the traditional inference rules. These similarities are illustrated in Table 5-5.

A close inspection of Table 5-5 shows that although there are significant similarities between the sets of properties that the network was sensitive to, there were also some quite significant differences. For example, in a number of cases, most notably with respect to type (i) Alternative and Disjunctive Syllogism problems, the network appears to employ what might be termed a 'default rule'. In addition, in the case of valid type (ii) Disjunctive Syllogism problems, L10 differentially classifies (in terms of bands) and processes problems, depending upon whether or not S1(V2) and S2 are negated. These results are without a doubt intriguing and deserve further consideration. They, and the conclusions which can be drawn from them, will be discussed in more detail in the next chapter.

One brief further clarification is appropriate with respect to the use of the term 'rule', as used to describe the properties to which the network is sensitive, in Table 5-5. This usage follows that employed by Berkeley *et al.* (1995). One of the things which made the use of the term 'rule' attractive to Berkeley *et al.* (1995) in their description of these sets of properties, was the fact that, despite there being some cases of differences between the features stipulated by traditional inference rules and the features to which L10 is sensitive, these differences appear to be comparatively minor. For example, whereas the traditional rule for Modus Ponens requires that the consequent be of the same sign as the conclusion, L10 is not sensitive to this feature. But because the training set is such that all instances of Modus Ponens problems in which S1(V1) is the same sign as S2 are also instances in which S1(V2) is the same sign as the conclusion, there is no necessity for the network to be also sensitive to this latter feature. Hence the difference between the features which L10 is sensitive to and those stipulated in the traditional rule of inference are trivial in this case. It is simply and artifact of the nature of the training set. An analogous and equally inconsequential difference arises in the case of Modus Tollens problems too.

There is a broad sense in which something is a 'rule' if it prescribes that, in circumstances X, behavior of type Y ought or will be indulged in by agent or system Z (Cf. the definition of 'rule' offered by Twining and Miers, 1976: p. 48). If 'rules' are understood in this way, then there should be nothing objectionable about the use of the term 'rule' in the context of Table 5-5, at least as a first approximation. The question of the status of these supposed network 'rules' will be addressed in more detail in the next chapter.

Conclusion

The results of analysis of the network L10 show, at least, that the objections and concerns about connectionist methodology which McCloskey raised in his discussion can be met, at the very least in principle. The information contained in the above tables and discussion serves to describe in detail the mechanisms responsible for the network's behavior. Moreover, Berkeley *et al.*'s (1995) discovery of rules within the network provides *prima facie* grounds for treating the claims of the advocates of the 'myth' which opened this chapter with at least a cautious skepticism.

It is also significant that, as a consequence of Berkeley *et al.'s* (1995) analysis of L10 there is now evidence about the operation of a trained network which can be used to assess whether or not networks can be said to have 'tokens', 'representations' and so on. Such notions are crucial to the CCTM, and the evidence from the analysis will be crucial in assessing the relationship between networks and the CCTM. This will also facilitate a realistic evidence-based assessment of the extent to which connectionist networks really do offer a challenge to the CCTM. The final outcome on these matters will depend upon an assessment of the network L10 with respect to the notions of 'systematicity' and 'compositionality' and a number of associated notions that are crucially related to the CCTM, which will be undertaken in the next chapter.

VI

Connectionist Networks and The Classical Computational Theory of Mind

Introduction

The last chapter began by raising some concerns about what was known about the functioning of trained connectionist networks. In particular, an epistemological difficulty with respect to the details of the operation of such systems after training was highlighted. This epistemological difficulty, it was argued, not only had consequences for the potential contribution that connectionist systems could make to cognitive theorizing in general, but also presented special difficulties for assessing the relationship between connectionist systems and the CCTM. The bulk of the chapter was taken up by the detailed description of the analytic method of Berkeley *et al.* (1995) which offered a means of resolving the epistemological problem and the results which they obtained from a particular network, L10.

With the information from this network in hand, the comparison between the results from at least one connectionist network (L10) and crucial properties, which have bearing upon the CCTM, is now tenable. Considering the results from the analysis of the network L10 and locating the significance of those results in the context of recent debates on CCTM related topics will be the main goal of this chapter.

The CCTM again

In the Chapter II, a discussion of the CCTM was presented to try to clarify this position. As a result of this clarification, a list of properties was generated. As a result of this clarification, it was apparent that the CCTM seemed to involve a commitment to the notion that CCTM computers (e.g. Turing machines) and minds share the following properties:

(1) A finite range of discrete tokens,
(2) A capacity to store and retrieve sequences of tokens,
(3) A capacity to perform a determinate range of precise and exceptionless operations upon tokens,
(4) A capacity to support a principled distinction between tokens and operations which manipulate those tokens,
(5) A capacity to construct structurally complex strings of tokens,
(6) A capacity to differentially perform operations upon structurally complex strings of tokens, dependent upon the order of the tokens in the string,
(7) A capacity to support principled interpretations.

These properties are closely related to one another and concern a variety of interrelated notions and issues. However, there is one particular pair of issues have come to the fore in recent years, with respect to the relationship between connectionist systems and the CCTM. These concern whether of not connectionist systems can exhibit the properties of systematicity and compositionality. The claims of Fodor and Pylyshyn (1988), and more recently Fodor and McLaughlin (1990) about connectionist systems with respect to compositionality and systematicity, have generated a substantial controversy. They claim that one of the principled differences between connectionist systems and what they (Fodor and Pylyshyn 1988: p. 12-13, *passim*) call 'classical' ones (i.e. those which are obviously consistent with the CCTM and also have the property mentioned in chapter 2, fn. 12) is that classical systems have these properties, whilst connectionist ones do not. Connectionists (e.g. Chalmers, 1990, Smolensky, 1990 and Pollack 1990) have made substantial efforts to refute this claim.

The challenge posed here directly relates to the properties associated with the CCTM, as it is in virtue of having of these properties, it is claimed, that CCTM machines exhibit compositionality and systematicity. In particular, Fodor and Pylyshyn (1988) maintain that having discrete tokens (i.e. property (1)) and a set of exceptionless operations defined for those tokens (i.e. property (3)) in combination, is sufficient to produce systematicity and compositionality. This point will be discussed further below. The issue of whether or not the network L10 is compositional and systematic consequently provides a natural starting place to begin considering the relationship between networks and the CCTM.

Systematicity

Fodor and Pylyshyn introduce systematicity in the context of linguistic capacities (1988: p. 37). However, they do not believe that the phenomenon is limited to the linguistic domain. They say (1988: p. 37),

> What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.

What they have in mind here is that the ability of someone to produce/understand the sentence 'John loves the girl', is intrinsically related to the ability to produce/understand the sentence 'The girl loves John'. Their claim is that one cannot produce/understand one without also being able to produce/understand the other.

However, one of the major problems with Fodor and Pylyshyn's (1988) critique is that it is far from clear exactly what they have in mind by the term 'systematicity' (See van

Gelder & Niklasson 1994 and Niklasson & van Gelder 1994). Niklasson and van Gelder (1994: pp. 288-289) note that,

> In their 1988 paper Fodor and Pylyshyn discussed systematicity at length, but provided no succinct and precise characterization of it; at best, they gestured at the phenomenon with hints, analogies and anecdotal observations.

In addition, despite the fact that Fodor and Pylyshyn (1988) claim that their systematicity argument is a 'traditional' one, they do not supply any other examples of it and (according to Niklasson and van Gelder 1994: p. 289) "...there is [almost] no occurrence of the argument or the concept in the cognitive science literature before 1988,...". This has presented some very real difficulties with respect to assessing whether or not certain connectionist systems and techniques actually serve as refutations of Fodor and Pylyshyn's claims (e.g. Chalmers 1990 and Smolensky, 1990).

Hadley (1994) has argued that in order to properly understand the relationship between Fodor and Pylyshyn's (1988) claims and connectionist counter-claims and the relationships of both sets of claims with respect to human performance, it is necessary to distinguish various degrees, or kinds of systematicity. In order to assess the network L10's systematic capacities, it will be helpful to adopt Hadley's framework. Hadley (1994) proposes that there are three distinct degrees of systematicity.

Hadley's (1994) descriptions of his three degrees of systematicity are couched in terms of linguistic items (i.e. 'words', 'sentences' and so on). However, Hadley's conceptions can be quite natural extended by employing more general terminology (e.g. 'tokens'). Doing this serves to make the conceptions of systematicity more straightforwardly applicable to networks such as L10, which are not trained upon linguistic tasks. This move is entirely consistent with the claims made about systematicity by Fodor and Pylyshyn (1988). With the appropriate substitutions made, Hadley's (1994) three conceptions of systematicity are as follows:

1) *Weak Systematicity*. A system is said to be weakly systematic if it can successfully process a novel set of strings of tokens which:
    - (i) contains no tokens that are not present in the set of strings of tokens that the system was trained upon, and
    - (ii) that no token in the novel set of strings of tokens is in a position within a string of tokens that it did not occur in (at some point) in the set of strings of tokens that the system was trained upon.

2) *Quasi-Systematicity*. A system is said to be quasi-systematic if:
    - (i) it can exhibit weak systematicity, and
    - (ii) the system can successfully process novel strings of tokens which are such that they contain embedded strings of tokens, and
    - (iii) both the novel embedded strings of tokens and the larger containing strings of tokens are (respectively) structurally isomorphic to strings of tokens in the set of strings of tokens that the system was trained upon, and
    - (iv) for each successfully processed novel string of tokens that contains a token in an embedded string of tokens, there is some string of tokens in the training corpus which does not contain embedded strings of tokens which contains the same token in the same position as it occurs in the embedded sentence.

3) *Strong Systematicity*. A system is said to be strongly systematic if:
    - (i) it can exhibit weak systematicity, and
    - (ii) the system can successfully process novel strings of tokens (both with and without embedding) which are such that they contain tokens that are in positions that those tokens did not appear in the set of strings of tokens which the system was trained upon, and
    - (iii) the novel set of strings of tokens which the system successfully processes has a significant fraction of tokens in positions that those tokens did not appear in in the set of strings of tokens which the system was trained upon.

What is the relationship between the seven properties associated with the CCTM and the notions invoked in the specification of the three kinds of systematicity?

Consider for example, the fact that weak systematicity presupposes some distinction between tokens and strings of tokens which is mirrored by properties (1) and (5) of the CCTM. The notion of weak systematicity also requires that strings of tokens be discrete from one another (as is indicated by the use of the term 'set'). It is unclear the extent to which this kind of systematicity will presuppose property (2) of the CCTM, because this will depend on the exact way that a particular systems is set up. Something like property (3) of the CCTM is also presupposed by weak systematicity, although it appears to require fewer (explicit) restrictions upon the nature of the operations than does the CCTM. Similarly, although the notion of weak systematicity presupposes that there is *some* distinction between tokens and the operations which manipulate those tokens (i.e. property (4) of the CCTM), but it does not necessarily require that the distinction is principled. Although some property broadly similar to property (6) of CCTM will be presupposed by weak systematicity, it is not clear the extent to which such a property would have to be identical to it. Finally, property (7) is only weakly required, in the sense that there is no reason why the interpretation needs to be principled, provided that there is *some* interpretation. So, despite the fact that there is an overall similarity between the requirements for weak systematicity and the properties associated with the CCTM, the requirements are considerably loosened, in a number of respects.

Quasi-systematicity has a condition that a system be weakly systematic. Consequently, it will presuppose at least the same properties of the CCTM, to the same degree as does weak systematicity. Notice though that there are some differences with respect to the properties of the CCTM. In particular, a system which lacks property (6) of the CCTM will not be able to handle the embedding. Thus, in the case of quasi-systematicity, property (6) becomes a requirement. Although the operations of property (3) of the CCTM may not necessarily have to be exactly as specified in the CCTM, it seems highly likely that systems which have exactly property (3) of the CCTM will be able to exhibit quasi-systematicity. In the case of property (7) of the CCTM there would be an analogous situation. Whilst it would certainly make quasi-systematicity easier to achieve if a principled interpretation could be supplied, there is no way to rule out the possibility that there may be non-principled means of achieving this.

In the case of strong systematicity, it would appear likely that exactly the seven properties of the CCTM would be required. Of course, it may turn out to be the case that some of the properties listed are not actually required. However, the precise issue with respect to connectionism verses the CCTM turns upon whether or not the properties of the CCTM are *necessary* for strong systematicity or not.

It is clear from the remarks of Fodor and Pylyshyn (1988) that what they have in mind when they talk about 'classical models', are models of the kind which exhibit the properties associated with the CCTM.[1] Indeed, Fodor and Pylyshyn's (1988) central claim is that, systematicity and compositionality emerge in classical models precisely because such models exhibit the properties associated with the CCTM. This being the case, it is appropriate to consider the network L10 with respect to systematicity and (given the relation between the two notions) compositionality in order to see what, if anything, the network can add to the debate.

---

[1] Fodor and Pylyshyn (1988) define what they take to be the two central properties of classical systems (constituent structure and structure sensitive processing) with respect to internal mental representations, as opposed to the external strings of tokens which were focussed upon in the discussion of the CCTM in chapter 2. However, it is clear from Fodor and Pylyshyn's (1988) discussion that they also believe that internal representations have properties analogous to those possessed by strings. That is, their view is very much a syntactic theory of mind. So, their overall view seems to be that this additional complexity is to be accomodated by attributing the possession of constituent structure and structure-sensitive processing to *internal* strings of tokens.

In the previous chapter, the position was advanced, based on the claims of Berkeley *et al.* (1995), that the network L10 had, during training, developed 'rules' for solving the Bechtel and Abrahamsen logic problem set. In a number of cases, the rules were also significantly similar to the rules of inference of 'classical' natural deduction systems. This then would seem as likely a level at which to find systematicity within the network as any other. Moreover, working at this level also will provide some insights into the similarities and differences between the network's rules and 'classical' ones.

The nature of the rules which Berkeley *et al.* (1995) discovered in their network L10 are such that, provided that novel inputs to the network can be represented without making any fundamental changes to the representational conventions at the input layer, then the network will be able to successfully process strings of tokens which contain novel tokens. A 'fundamental change' here would be, for example, to increase the dimensionality of the combinations of units used to represent the letters at the input layer (for example, by encoding letters across three units rather than just two). The nature of the properties to which the network is sensitive to at the hidden layer guarantee this capacity of the network . What the network detects at the hidden layer is not the presence or absence of particular letters in particular positions at the input layer, but rather the relationship between variables at the input layer. Thus, subject to the constraint that encoding novel tokens does not involve any fundamental change at the input layer, there is every reason to believe that the network would be able to successfully process problem containing entirely novel tokens.

This evidence suggests that the rules of inference discovered in the network by Berkeley *et al.* (1995) give rise to behavior which is at least weakly systematic. Indeed, if we consider closely the properties that the network is sensitive to in Modus Ponens problems for example, the reasons for this weak systematicity can be seen. Consider the properties listed in Table 6-1.

PROBLEM TYPE	FORMAL DEFINITION OF RULE FOR VALID PROBLEM TYPE	NETWORK 'RULE' IDENTIFIED BY THE INTERPRETATION
Valid Modus Ponens (MP)	S1(V1) = S2 S1(V2) = C SIGN S1(V1) = SIGN S2 SIGN S1(V2) = SIGN C CONNECTIVE: IF...THEN	S1(V1) = S2 S1(V2) = C SIGN S1(V1) = SIGN S2 CONNECTIVE: IF...THEN

Table 6-1
Network rule for Modus Ponens, as compared to the traditional rule.

The rule only specifies relations between variable letters and signs of those variables, in addition to the fact that the main connective must be IF...THEN.... If part of the training set had been held back from the network during training and the network had developed the same rule, then under the right conditions, the network would have been able to 'successfully process' the withheld portion of the training set. For example, if the withheld portion of the training set satisfied conditions (i) and (ii) for weak systematicity, then the network may well exhibit weak systematicity. This is guaranteed by the network's inference rule. An analogous situation obtains with the rule for Modus Tollens and the networks other rules (except for the default rules, which will be discussed momentarily). So, on the basis of the evidence from the network's rules, it is reasonable conclude that the network is weakly systematic.

Although there are definite similarities between the networks rules of inference and traditional inference rules though, there are also important differences. The network is sensitive to (almost) the same properties as the traditional rules for Modus Ponens, Modus Tollens and so on, as specified by the requirement that there be a specific connective and as a set of specific relations between variables and signs. However despite these facts, the manner in which the connective and the variables and signs are available to the network is not the same as it is in the more traditional case.

There is a further presupposition, in the case of the formal definition of the traditional rules of inference, which the network does not share. This is the presupposition that either the various properties which the rule is sensitive to will be tokened in a manner such that the tokens are either themselves primitive, in so much as that they can be assigned principled (i.e. non-disjunctive) interpretations, or are constituted (in a legal manner) out of strings of tokens which are primitive in the required sense. Nonetheless, for the problems which the network was trained upon, the network's rules suffices to produce the appropriate behavior. That is to say, it 'successfully processes' the problems. Moreover, provided that novel problems can be presented to the network in a manner which does not require a fundamentally change (in the sense described above) to the representational conventions, then the network will continue (for the most part) to successfully process such problems. However, this constraint is fundamentally different from that supposed in the classical case. In addition, it may well present some very real difficulties when it comes to the generality (with respect to novel problem types) for the network.

In order for the network to be able to successfully process a Modus Ponens problem, for example, such as $((A \supset B) \supset A), A \supset B$, therefore A, it would be necessary to find some means of representing such a problem without making a fundamental change in the representational conventions of the network. Now, although it may be possible to develop some convention which can satisfy this requirement, it is not obvious how best to proceed. In the case of the traditional formal rule, by contrast, the presupposition about the nature of the tokening of the components of the problem is such that it provides an unambiguous manner in which to proceed.

These difficulties come to the fore especially strongly in the case of the so-called 'default rules' developed by the network. Consider the network's rule for valid type (i) Alternative Syllogism problems, as presented in Table 6-2.

PROBLEM TYPE	FORMAL DEFINITION OF RULE FOR VALID PROBLEM TYPE	NETWORK 'RULE' IDENTIFIED BY THE INTERPRETATION
Valid Alternative Syllogism (AS) (i)	S1(V1) = S2 S1(V2) = C SIGN S1(V1) ≠ SIGN S2 SIGN S1(V2) = SIGN C CONNECTIVE: OR	CONNECTIVE: OR

Table 6-2
Network rule for type (i) Alternative Syllogism, as compared to the traditional rule.

In the case of problems of this type, the network is only explicitly sensitive to the fact that the main connective of the problem is OR. It is reasonable to assume though that the network also presumably makes use of the fact that there are none of the other properties to which it is sensitive, in successfully processing problems of this type from the training set. That is to say, the network trades upon the idiosyncratic properties of the input encodings, in conjunction with some general properties of the problem set as a whole, to actually solve problems of this kind. However, once again for the network to successfully process novel problems which are more complex, a means would have to be found of presenting these problems to the network in a manner which at least did not require a fundamental change in representational conventions.

Despite these apparently negative diagnoses of the network's abilities, the rules recovered from the network L10, suggest that the network could be weakly systematic in Hadley's (1994) sense, under the appropriate circumstances. In addition, other versions of the network exhibited good generalization when trained upon fractions of the complete training set and tested on the remainder. Results reported in Dawson *et al.* (1997) suggest that the system may even exceed Hadley's (1994) requirements for a system to being weakly systematic. Dawson *et al.* (1997) report a pilot study in which they trained a value unit network in the same manner as L10. They then presented a set of problems to the network that was identical to the training set except for the fact that in every instance where there was a '0' in a variable position in the original training set, the '0' was changed to 0.25, and where there was a '1' in a variable position in the original training set the '1' was changed to 0.75. The network's performance on these problems was extremely good. In 94.3% of cases, the network gave an appropriate response (i.e. successfully processed) to the problems.

It must be admitted, however, that these results are not unequivocal. Condition (i) for a system being weakly systematic requires that the novel set of strings of tokens contains no tokens that are not present in the set of tokens that the system was trained upon. Dawson *et al.*'s pilot study involved presenting the network with a set of problems which were constructed out of activation values the network had not seen before. On the one hand, these activation values could be taken as representing novel tokens. On the other hand though, they could be taken as merely testing the ability of the network to handle degraded input (i.e. exhibit graceful degradation). Only on the first reading would this result be suggestive of more than weak systematicity. This is because (on this reading) although the tokens were similar in kind to the ones which the network was trained upon, and as such did not require a fundamental change in representational conventions, the network had not been presented with exactly these tokens during training. A more detailed investigation of this phenomenon would be an interesting line of future investigation.[2]

Although it may not sound like too much of an achievement to produce a network which is merely weakly systematic, Hadley (1994: p. 14) notes that, "Even the ability to demonstrate weak systematicity is no small feat." In fact, elsewhere Hadley and Hayward (1997: p. 5) suggest that it is possible that even weakly systematic systems may be sufficient to refute Fodor and Pylyshyn's (1988) claims.[3] Thus, the fact that systems such as L10 may exhibit this kind of systematicity is far from trivial, as it provides the beginning of a response to Fodor and Pylyshyn's objections to the connectionist research program. This is the positive conclusion which can be drawn from the network L10. However, there is also a less positive conclusion which can be drawn from the network.

---

[2] On this reading, the network's rules might be interpreted as enabling it to exhibit what Niklasson and van Gelder (1994) call 'Level 3' systematicity.

[3] Hadley (1994) assesses Chalmer's (1990) network to be weakly systematic. In Hadley and Hayward (1997: p. 5) it is stated that "...we believe that the work of Chalmer's (1990) and Smolensky (1990) may very well constitute counterexamples to F&P's general claims." Similarly, it is also noted that "...Smolensky's and Chalmers' results may separately refute F&P,...".

It should also be clear that the network L10 will not be able to exhibit either strong systematicity or quasi-systematicity. The reasons for this are closely related to the differences between the presuppositions made about tokens in formal inference rules and the tokens operated upon by the networks rules. Both strong and quasi-systematicity make appeal to the notion of 'embedding'. However, in order for the rules of the network to be able to handle embedding, significant and substantial extensions to the basic structure of the model would be required.[4] This is because, as things stand, there is no obvious means by which such embedding could be executed within the network L10, in a manner which will guarantee that the networks performance will not degrade.

The fact that the rules in the network L10 are such that they can only support weakly systematicity, also will have consequences, with respect to the networks ability to exhibit compositionality. This is because of manner in which Fodor and Pylyshyn (1988) link the two notions. It is now appropriate to consider these consequences.

Compositionality

Fodor and Pylyshyn (1988: p. 42) describe what they term the "Principle of Compositionality" as follows;

---

[4] Perhaps for example, if features similar to those found in Pollack's (1990) RAAM architecture were incorporated into value unit networks, then such network could exhibit higher degrees of systematicity.

> ...insofar as a language is systematic, a lexical item must make approximately the same semantic contribution to each expression in which it occurs.

It is not unreasonable though to question the precision of Fodor and Pylyshyn's (1988) notion of compositionality, in the light of the problems which have been demonstrated with their notion of systematicity, particularly given that the treatment of topics within their paper is remarkably uniform. This being the case there are grounds for believing that the notion of compositionality may also be in need of some conceptual clarification, in a manner analogous to that done by Hadley (1994) for systematicity. In addition, the evidence from the analysis of L10 by Berkeley *et al.* (1995) provides a context in which the relevant questions can be addressed.

One point which needs to be addressed at the very outset though is Fodor and Pylyshyn's (1988) terminology.[5] When Fodor and Pylyshyn use the term 'compositional' they seem to have in mind circumstances in which individual constituents of a complex expression make (approximately) the same contribution to all expressions in which they occur. This usage of the term is not standard however. It is far more common (van Gelder 1990: p. 356, fn. 1) for the term 'compositional' to be used for circumstances which Fodor and Pylyshyn call 'combinatorial'. Roughly speaking, combinatoriality (in the Fodor and Pylyshyn's sense) is a property of a language or representational scheme whereby the meaning of complex expressions is a function of the meanings of the simpler parts which make up the complex expression. For the sake of consistency, in what follows, Fodor and Pylyshyn's terminology will be employed. So, the crucial issue is whether or not individual constituents of a complex expression make (approximately) the same contribution to all expressions in which they occur in the rules of inference of the network L10.

---

[5] The confusion over terminology noted here may also serve as further evidence that there is a lack of clarity and precision, analogous to that which arises with 'systematicity', with respect to Fodor and Pylyshyn's (1988) notion of 'compositionality'.

However, a further question arises in this context. The question concerns whether or not the network L10 can be said to have 'tokens' at all, in the appropriate sense. There is an argument which suggest that, at least in some sense, the network will have tokens. Any system which processes information must have some means by which information can be input into the system. Given that connectionist networks are reasonably unambiguously information processing systems, it follows that connectionist networks must have some mechanism for inputting information into them. As the role of 'tokens' (at least in the CCTM) is to be the bearers of information, it follows that, there must be something which is at the very least strongly analogous to tokens within the network L10. There is a problem though.

In a Turing machine, an individual token plays a very precise role. This role is defined by the machine table. The same is not straightforwardly the case in the network L10. This is because of the employment of distributed encodings within the network. For example, input units one and two are used to encode the variable letter in the first premise. A result of this is that, if input unit one has the value of 1, then it is not clear what it is contributing to the networks input. If input unit two is set to 0, then it could be indicating that the input variable S1(V1) is a 'B'. If input unit two is set to 1, then it could be indicating that the input variable S1(V1) is a 'C'. Although individual units may be such that they can be subject to interpretation, they cannot be interpreted in a *principled* manner. That is to say, they do not make the same contribution to the network under all circumstances. This is because they cannot be assigned a single interpretation which can be coherently applied in all cases. This is obvious from the disjunctive nature of the interpretation of, for example, input unit one. If input unit one has a value of 1, then the input variable is either a 'B' or a 'C'. If input unit one has a value of 0, then the input variable is either a 'A' or a 'D'. This suggests that individual processing units are not tokens, in the all the respects relevant to the CCTM, although there are parts of the network which function 'as if' they were tokens (i.e. the components which act as inputs to the network's inference rules).

It is clear that in order for a system to exhibit compositionality in Fodor and Pylyshyn's (1988) sense, it must support some distinction between simple and complex tokens. The lack of any such distinction in networks is one of the main grounds Fodor and Pylyshyn offer for why connectionist systems cannot have this feature. Notice though, that the contrast between simple and complex expressions is ambiguous. On the one hand, it could refer to the contrast between individual tokens verses strings of tokens. On the other hand, it could refer to the contrast between strings of tokens which do not contain any embedding, verses those which do contain embedding. For example, a single token, say an 'A' would be simple in both senses, whereas 'A $\supset$ B' would be complex in the first sense, but not in the second. An expression like '(A $\supset$ B) $\supset$ A' though would be complex in both senses.

The differences between these two distinctions can be captured by thinking in terms of the 'degree' of nesting of tokens, in a manner analogous to that sometimes employed in the study of certain logical properties (see Bermann, Moor and Nelson 1990: Ch. 6). A simple 'A' can be thought of as a formula (i.e. an expression) of degree 0, as it contains a single token. The expression '$A \supset B$' can be though of as being of degree 1, as it contains more than one token. In addition, although this expression contains a component (to wit, the '$\supset$') which can potentially support embedding, it does not actually do so. The expression '$(A \supset B) \supset A$' would count as being of degree 2 as it contains one instance of embedding. The notion of degrees can be extended so as to capture arbitrary degrees of complexity, with respect to embedding. The purpose of introducing the notion of degrees here is that it enables the requirement of complexity, as employed by compositionality (in Fodor and Pylyshyn's sense), to be specified in a concise and non-ambiguous manner. In addition, it makes an assessment of the compositionality which the entities upon which the rules of the network L10 operate upon possible.

It seems likely that in order for a system to be entirely compositional in the sense intended by Fodor and Pylyshyn, it will be necessary that individual constituents of a complex expression make (approximately) the same contribution to all expressions irrespective of the degree of embedding in an expression. This, it seems, is the strongest type of compositionality possible. For this reason, it might be termed 'strong compositionality'. The choice on nomenclature here is intended to suggest an analogy with Hadley's (1994) notion of strong systematicity.

It is clear though that the rules of the network L10 do not exhibit strong compositionality. The reason this is the case is that the network has no (obvious) means of handling embedding. This, after all, was the reason why the network failed to be able to exhibit more than weak systematicity. Notice though, that with the newly introduced terminology for degrees of embedding, we can specify with greater precision the exact extent of the networks deficiency. In order for a network to exhibit embedding of the kind presupposed by Hadley's (1994) notions of strong and quasi-systematicity, what is required is embedding greater than of degree 1. This is just what the network failed to be able to do. The network was nonetheless able to exhibit embedding of up to degree 1.

There seems to be a sense in which the rules of the network were able to exhibit something akin to compositionality. After all, the rules were able to solve the Bechtel and Abrahamsen logic problem shows that to some degree, constituents of a complex expression must have made (approximately) the same contribution to all expressions in which they arose. If this were not the case, then the network would not have been able to distinguish between valid cases of Modus Tollens problems and invalid cases of Modus Ponens problems which involve affirming the consequent. This suggests that there may be a sense in which it is legitimate to say that the network exhibited what might be termed 'weak compositionality'. Once again, the similarity with Hadley's terminology is intended. A system is weakly compositional in this sense, if individual constituents of complex expressions (i.e. expressions with embedding of a degree greater than 0) make (approximately) the same contribution to all expressions with embedding up to, but not greater than, degree 1. So, this being the case, the rules of the network L10 not only were capable of exhibiting weak systematicity, they also exhibited weak compositionality.

Given the nature of the training set that the network L10 was trained upon, it is no great surprise that it did not develop rules which were more than weakly compositional. This is because the Bechtel and Abrahamsen logic problem does not contain any problems with embedding greater than degree 1. It is reasonable to wonder then what would happen if a slightly more complex problem was studied. For example, what if the training set were expanded so that it contained problems with embedding of degree 2? Suppose a network were to be able to successfully learn to solve all the problems of such an expanded training set and be able to successfully process (in Hadley's sense) novel strings which included embedding to degree 2, but no greater degree. In such a case, the network would clearly not be strongly compositional. On the other hand, it would be more than just weakly compositional. Such a case seems to provide grounds for positing a third kind of compositionality, which I will call, again following Hadley's (1994) example, 'quasi-compositionality'. However, it is clear that the procedure used for specifying kinds of compositionality used in the previous two cases cannot be employed here, unless we want to say that there is a different kind of compositionality for each degree of embedding that a system can handle. Clearly a more general approach must be found to 'fill in the gap', so to speak between weak and strong compositionality. One approach would be to say that a system can be said to exhibit quasi-compositionality if individual constituents of complex expressions (i.e. expressions with embedding of a degree greater than 0) make (approximately) the same contribution to all expressions with embedding up to, but not greater than, degree *n*, where *n* is the highest degree of embedding found in the training set of that system. Notice that a condition on any system exhibiting quasi-compositionality will be that it is weakly compositional. However, such a system would not be strongly compositional, unless individual constituents of complex expressions made (approximately) the same contribution to all expressions with embedding up to degree n+1 or greater.

Notice the strong analogy which exists between the three kinds of compositionality and Hadley's (1994) three kinds of systematicity. To some extent, this is intentional, however it is also just what one would expect if the two notions are very closely related to one another or, as Fodor and Pylyshyn put it (1988: p. 41), "...aspects of a single phenomena." Further evidence of the affinity between the two (sets of) notions comes from the fact that it is exactly the same deficiency, the inability to handle strings of tokens with embedding (greater than degree 1), which causes the rules of the network L10 to be only weakly systematic, which also causes them to be only weakly compositional. Moreover, traditional rules of inference are strongly compositional, just because they presuppose no limit to the degree of embedding, and this in turn explains why they can give rise to strong systematicity.

What then, are we to infer from all this about the relationship between the network L10 and the CCTM? At the beginning of this chapter a brief discussion of the relationship between the CCTM and Hadley's various notions of the systematicity was presented. Overall, properties broadly similar to those of the CCTM were required, although the requirements upon those properties appeared to be far less stringent than was the case for strong systematicity. Beyond this though, it is hard to be more specific. Not least because of the limited nature of the evidence which could be gained from the network L10.

The goal of Berkeley *et* al. (1995) was just to investigate the interpretability of trained networks and not consider issues such as systematicity and compositionality. Rather, the unanticipated conclusions about the status of systematicity and compositionality, with respect to the network, are significant benefits which comes from the network having been interpreted. Although the strength of the conclusions which can be drawn in the current context are limited to some extent, they nonetheless indicate that a potentially fruitful direction of research would be to develop and analyze networks which could handle embedding of various degrees. Such networks should be able to exhibit quasi-compositionality and quasi-systematicity at the very least. This would produce considerably more evidence about the exact relation between networks and the CCTM. The development of connectionist networks which could exhibit strong compositionality and consequently be strongly systematic would be the ultimate goal though.[6]

Of course, the extent to which such networks exhibited these features would only be one element in developing a full response to Fodor and Pylyshyn (1988). In fact, their arguments suggest that they think it highly likely that such networks could be developed. This is because they do not believe that, as a matter of principle, that connectionist systems cannot exhibit systematicity and compositionality. What they *do* believe though is that any system which has these properties will just be an implementational variant of what they call a 'classical system'. Classical systems, in this sense, are those which have just the properties associated with the CCTM. Whether or not there might be systems which could be strongly compositional and systematic, but which were not merely implementational variants of systems with all the properties associated with the CCTM has yet to be shown. However, the evidence from the analysis of L10 suggests that the conclusion (that such systems are not possible) is less simple and straightforward than Fodor and Pylyshyn suppose.[7] This matter will be addressed in the next section.

---

[6] A network which exhibits strong systematicity has recently been described by Hadley and Hayward (1997). However, they are cautious to say that, as the network is not purely connectionist (in Fodor and Pylyshyn's sense), their network does not provide a clear refutation of Fodor and Pylyshyn's (1988) claims.

Rules and Cognitive Systems[8]

Let us begin by sketching a little of the theoretical context in which Fodor and Pylyshyn's (1988) challenge arises. One of the intuitions which originally motivated the adoption of the CCTM within cognitive science in general, and the philosophy of mind in particular, was the intuition that cognition is information processing. One of the goals shared by both these fields is the goal of being able to express useful generalizations about the mind and cognitive functioning. However, it is generally agreed that in order to be able to offer useful generalizations about cognitive functioning, it is necessary to draw a distinction (at the very least) between the implementational level and the level at which the functional architecture of a system is described. The reason this is the case is the fact that two systems can do (roughly) the same thing computationally, whilst having precious few physical properties in common. For example, both the average human being and the average electronic calculator can add and multiply, despite the fact that the one is made up of biological matter, whilst the other is constructed out of silicon. Such differences exist, so the standard view goes (see for example Fodor 1975, Marr 1982 or Pylyshyn

---

[7] I do not intend to imply by this remark that any connectionist system for which a GOFAI variant has not been proposed will count as not being an implementational variant. See Dawson, Medler and Berkeley (1997), for further details on this point.

[8] Part of the argument in this section is based upon that presented in Dawson, Medler and Berkeley (1997).

1984), only at the implementational level. That is to say, at the level of the physical substrate upon which each function is instantiated. By contrast, the regularity of two systems performing multiplication is something which has to be captured at a higher level.

The distinction between these two levels is important, in so much as there can be changes at the implementational level which have no effect upon the state of a system as described at a higher level. For example, both the multiplying human and the electronic calculator could be heated up (within certain parameters) and this might have no effect whatsoever upon the fact that both systems performed multiplication. This is not to say though that the implementational level is of no consequence whatsoever for higher levels. The level of the functional architecture (i.e. that above the implementational level) must be such that there is some primitive set of information processing capacities which can be given functional descriptions and which are ultimately explained by appealing to natural laws, which operate at the implementational level. In order to be truly useful with respect to cognitive theorizing though, a system must be such that it can also be given a description at a third level. This is what Pylyshyn (1984) terms the semantic level.

In the current context, the crucial distinction is between the implementational level and those above it. The reason for this is that the implementational level plays no direct role in cognitive theorizing. So, in Pylyshyn's view, any system which is merely an implementational variant of another system, will have nothing new to add to cognitive science. This is what gives Fodor and Pylyshyn's objections to the connectionist research project their teeth. If they are correct, then no matter what properties connectionist systems exhibits, all such a system will be able to provide are non-cognitive implementational level accounts. However, if connectionism is to be informative to cognitive science (or the philosophy of mind, for that matter), it will be necessary that networks can give rise to accounts which are at a higher level, namely the cognitive one.

What then, it is reasonable to ask, is the cognitive level exactly? Pylyshyn (1991: p. 191) answers this question when he states that,

> The cognitive architecture...is the level at which the states (datastructures) being processed receive a cognitive interpretation. To put it another way, it is the level at which the system is representational...

Similarly, what is it exactly which distinguishes a cognitive theory from an implementational one? Again, Pylyshyn (1991: p. 191) gives an answer when he remarks that,

> Notice that there may be many other levels of a systems organization below this [the level of the cognitive architecture], but these do not constitute different cognitive architectures, because their states do not represent cognitive contents. Rather, they correspond to various kinds of implementations, perhaps at the level of some abstract neurology, which realize (or implement) the cognitive architecture.

However, if this is the case, then the analysis performed by Berkeley *et al.* (1995) seems to provide good grounds for maintaining that the network *was* cognitive in the relevant sense. The crucial distinction which divides the genuinely cognitive from the implementational is the issue of whether or not a system is 'representational', or has states which 'represent cognitive contents'. However, the analysis of L10 showed that this was just the situation within the network, as it demonstrated that particular network states (i.e. levels of activity in the hidden units) could be associated with particular semantic interpretations.

This being the case, the analysis of the network L10 can be seen as providing evidence, of sorts, for cognitive connectionist networks. The fact that the network was also weakly systematic (in Hadley's 1994) sense and also was what I termed 'weakly compositional', suggests that the situation with networks may be less clear cut than Fodor and Pylyshyn (1988) would have us believe.

The network L10 succeeded in coming up with an entirely novel theory of how to solve the Bechtel and Abrahamsen (1991) logic problem set. This theory is embodied in the seven inference rules discovered by Berkeley *et al.* (1995). This theory provides further evidence that the network is not just an implementational level account, in two respects. First, the seven rules succeed in capturing generalizations about the problem set. This is just the kind of thing that is required of a theory at the cognitive level. Second, the nature of the theory developed by the network is such that, at least in principle, it would be possible to run psychological experiments on human subjects to determine whether such subjects solved the problems in an analogous manner.[9] For example, it would be interesting to see whether humans processed valid disjunctive syllogism problems with negated second premises, in a manner different from those with non-negated second premises.

---

[9] For example, subjects could be trained to solve the problem by being given only information about their errors, similar to the back propogation proceedure used to train the network. The subjects could then be asked to report how they solved the various kinds of problems in the training set and how their solutions compared to the the way the network solved problems of the same kind.

A brief word of caution is in order here though. In no way is any of the above intended to imply that the network L10 *actually had* cognitive states. The point is rather that, contrary to the claims of Fodor and Pylyshyn (1988), the network is a theory at the cognitive level, albeit one which has limited plausibility. Showing that, in principle, networks can be used as a means of generating theories at the cognitive level, is one of the more significant conclusions which can be drawn from the analysis of network L10 (see Dawson *et al.* 1997).

The question of the status of the rules discovered in the network, has yet to be addressed. In the above it has been useful to consider the rules recovered from the network, in the light of Fodor and Pylyshyn's challenges with respect to systematicity and compositionality in networks. However, a question still remains about the nature of the rules themselves, as compared to the rules of a Turing machine. The question is, in what sense (if any) are these items rules in the classical sense of the term?

It would seem that there is a sense in which, for the network itself, the rules are not at all classical, outward appearances notwithstanding. What is crucial here is the issue of the tokens which are processed by the network in the course of the operation of the 'rules'. If it is taken to be the case that a crucial property of tokenhood in the classical sense is that the tokens have a fixed interpretation (or to use the terms employed earlier, that tokens have a non-disjunctive interpretation), then strictly speaking, there are no such items within the network. After all, all information within the network is captured by unit activations and, as was argued earlier, it is only in very rare and exceptional cases that units can be assigned non-disjunctive interpretations. However, if this is the case, what are we to make of the ‘rules’ of the network, especially their apparently cognitive nature just argued for?

The situation which arises within the network L10 seems to be perfectly captured by what Smolensky (1988) calls the ‘subsymbolic paradigm’. Smolensky (1988: p. 3) suggests that,

> ...cognitive descriptions [can be] built up of entities that correspond to *constituents* of the symbols used in the symbolic [i.e. CCTM] paradigm: these fine grained constituents could be called *subsymbols* and they are the activities of individual processing units in connectionist networks.

This seems to be exactly the situation within the network L10. The ‘rules’ discovered by Berkeley *et al.* (1995) operate over tokens (or ‘symbols’ in Smolensky’s terminology), but are not themselves instantiated in terms of tokens. This provides an explanation for why the network can be described as having come up with a novel cognitive theory for solving the Bechtel and Abrahamsen logic problem, yet does not itself appear to be a bearer of tokens in the required (classical sense). The symbolic descriptions which constitute the networks rules are just descriptions which map onto the underlying activity of the processing units, which takes place at the subsymbolic level.

However, it is important to realize the significance of the analysis of the network performed by Berkeley *et al.* (1995) in making this link between the levels explicit. Indeed, Smolensky (1988: p. 3) clearly acknowledges the importance of establishing this link when he remarks that,

> ...it is often important to *analyze* connectionist models at a higher level; to amalgamate, so to speak, the subsymbols into symbols.[10] (emphasis added)

Moreover, this view seems to be entirely consistent with that described by McCloskey in his critique of connectionist methodology. Without network analysis and interpretation, the mapping between subsymbolic operations and symbolic operations is left entirely occult.

What the analysis of the network L10 succeeds in showing is that, to at least some extent (i.e. to the extent to which weak systematicity and weak compositionality can be important as explanatory concepts within cognition), the subsymbolic operations of a network can be used to provide explanatory accounts of cognitive functioning at the symbolic level. Whilst this conclusion does not succeed in meeting the entire challenge to connectionism proposed by Fodor and Pylyshyn (1988), it does suggest that there are grounds for cautious optimism, with respect to the capacities of connectionist networks to provide explanations of cognitive phenomena, including those which involve compositionality and systematicity. Moreover, the evidence suggests that some of this work needs to be done at the conceptual level in order to get a clear idea of *exactly* what the notions of compositionality and systematicity involve. There is also work to be done in developing and, all importantly, analyzing more powerful and sophisticated networks.

Conclusion

As a matter of fact, it should be no great surprise that the network L10 did not provide a knock-down refutation to Fodor and Pylyshyn (1988). After all, this was not the goal which was being pursued when the network was developed. In fact, given the simplistic nature of the encoding scheme used with the network and the comparatively simple nature of the logic problems involved, it would have been more than surprising if the network could be used to refute such a position! However, what the network does succeed in doing is raising a number of important questions about the relationship between the CCTM and networks. More importantly, the fact that the network exhibits weak systematicity and the notion of weak systematicity seems to involve many notions which, though not identical to, are broadly similar in some respects to those invoked by the CCTM, shows that the relationship between networks and the position has yet to be fully understood. Until, for example the relationship between the symbolic and the subsymbolic is delineated clearly, it is too early to say that the final word on the matter has been said.

---

[10] Smolensky (personal communication, 1992) has confirmed and emphasized the importance he attaches to network analysis, as a means of bridging the gap between the symbolic and the subsymbolic.

It may be the case that the study of connectionist systems may make it necessary that certain familiar notions are sharpened or broadened. Indeed, one might take Hadley's (1994) work on systematicity as being a first step in this process. Prior to its application to networks by Fodor and Pylyshyn (1988), the notion of systematicity was employed without too many difficulties or controversies. Now, through the work of Hadley, what might be termed the 'open texture' (see Waismann 1951) has been revealed. I have also suggested in this chapter that the term 'compositionality' may similarly require further sub-division and refinement. However, the main 'take-home messages' from this chapter are that there is good evidence that connectionist systems have a positive contribution to make to cognitive science and the philosophy of mind, and that it is much too simplistic to just claim, as does Sterelny (1990: p. 168) for example, that

> Connectionists offer a rival view [to the CCTM] of the architecture of the mind, the nature of mental representation, and the nature of operations on those representations.

Similarly, it is much too soon to proclaim a new Kuhnian 'paradigm' (Schneider, 1987). There appear to be many subtle and complex relations which obtain between the CCTM and any putative alternative which might be proposed on the basis of connectionist models. Moreover, these relations are deserving of further concentrated study both within cognitive science and philosophy, as it appears that they are only just beginning to be understood.

VII

Conclusion: Connectionism, Present, Past and Future

Introduction: The Present

In the preceding Chapters, the position I have been calling the Classical Computational Theory of Mind (CCTM) was introduced. As the position lacked a clear formulation, an attempt was made to come up with something like the bare bones of the position by examing in detail the metaphor upon which it is based. By taking a Turing machine as the paradigm case of a computational device, it was possible to find a set of seven properties that appear to be shared by both minds and computers so conceived. This set of properties then provided a basis upon which an alleged alternative to the CCTM could be evaluated.

The particular alternative to the CCTM which has been the main concern here is what has come to be known as 'connectionist systems' or 'networks'. These systems, it is claimed by some (e.g. Searle and Dreyfus), appear to have just the kinds of properties needed to meet some of the objections which have been raised against the CCTM. It has also been claimed (e.g. Schneider, 1987) that the development of such systems represents a 'paradigm shift' (in the Kuhnian sense) within cognitive science, away from the CCTM. One of the primary goals here has been to determine whether or not this is really the case. This necessitated the introducing of connectionist systems in some detail. In particular, attention was focused upon the class of connectionist systems which undergo training using what is known as 'back propogation' style learning procedures. The reason for this focus is that this class of systems is the most widely discussed in the philosophical literature on the subject and because much of the philosophical excitement which has been generated by connectionist research stems in large part from the consideration of systems of this class.

Once some of the technical details of connectionist networks had been introduced, some of the claims which had been made about such systems were examined in detail. These claims concerned both the relationship of connectionist systems to those more obviously in the spirit of the CCTM, and the status of connectionist systems as models of cognitive functioning. These claims were called 'The Myths of Connectionism', because upon close examination it turned out that many of these claims were problematic. Some of the claims were just false, others were only true when significantly qualified in an appropriate manner. However, as much of the initial attractiveness of connectionism (especially as the basis of an alternative to the CCTM) derived from the myths, it then became reasonable to examine in some detail the real differences and similarities between connectionist systems and the CCTM.

One particularly important set of claims about connectionist systems concerned the nature of tokens and the operations upon tokens within connectionist systems. However, it was argued that there was a significant epistemological and methodological problem which had to be solved before such claims could be taken seriously. After networks had undergone training their complexity was such that there was no way to determine what was going on within them, unless they were subject to detailed analysis. Regrettably, the detailed analysis and interpretation of trained networks is not commonly undertaken. Without such analysis though, claims made about the nature of tokens and operations within networks lack adequate justification. Moreover, the lack of such analysis significantly undermines the status of trained networks as models of cognitive function. Given these facts, the only way to determine whether, and to what extent, networks provide an alternative to the CCTM was to consider in detail a network which had been subject to analysis and interpretation.

A technique for network analysis and interpretation, developed by Berkeley *et al.* (1995), was then introduced. It turns out that the results of applying this technique to a network trained upon a set of logic problems, originally studied by Bechtel and Abrahamsen (1991), gives the required detailed features of network functioning. Berkeley *et al.*'s (1995) interpretation of this network thus provides the necessary evidence upon which a close assessment of the relationship between the CCTM and networks can be based. Perhaps more significantly though, the network developed a number of features which seemed to run contrary to the claim that networks are a radical alternative to the CCTM.

When the features recovered from the interpretation of the logic network were compared with the set of seven properties associated with the CCTM, it turns out that although there are differences, the differences are not as radical as might be expected. In particular, the network appeared to have tokens and operations of a kind which were fundamentally similar in certain important respects to those associated with Turing machines (and minds), by the CCTM. This shows that, whist there may be some grounds for reconsidering some of the notions which are centrally associated with the CCTM, the tenability of the claim that connectionist research offers a 'radical alternative' to the position is undermined.

This much having been said, it may be the case that connectionist research might provide the basis for a reconception of some aspects of the CCTM. However, the extent to which this is the case has yet to be fully determined. Rather, the analysis of the logic network, in conjunction with other recent work (for example, that of Hadley 1994) shows that the unqualified claim that there is a sharp and radical difference between connectionism and the CCTM, typified for example by Fodor and Pylyshyn (1988), is just too simplistic. There appears to be significant conceptual complexity lurking just below the surface when familiar notions (such as 'systematicity' and 'compositionality') are considered in the light of connectionist research. Instead of supporting the conclusion that there is a radical alternative to the CCTM though, the correct conclusion appears to be that more work needs to be done to refine the constituent concepts of the CCTM and allied positions.

Before proceeding any further though, there is an outstanding historical matter, with respect to the relationship between networks and the CCTM, which deserves some attention. Although in recent years it has become a commonplace to contrast networks with the CCTM (albeit incorrectly), this is a comparatively new phenomenon. Despite the fact that some historical precedent for this opposition has been claimed, it can only be plausibly claimed on the basis of a revisionist view of history. In the next section, I will try and briefly make a case that something like the view I have been advancing here is in fact entirely consistent with the much of the history of Artificial Intelligence and cognitive science research (perhaps with the exception of the last decade or so). I will then turn my attention to the future by suggesting and briefly outlining some potentially fruitful lines of research which may serve to clarify further the relationship between the CCTM and networks. In addition, I will attempt to suggests the kinds of evidence, beyond that gained from the analysis of Berkeley *et al.*'s (1995) network, which will be required to develop a more through understanding of important concepts such as 'tokenhood', 'compositionality' and so on.

History: The Past

According to the standard (recent) history of connectionism (see for example the accounts offered by Hecht-Nielsen (1990: pp. 14-19) and Dreyfus and Dreyfus (1988), or Papert's (1988: pp. 3-4) somewhat whimsical description), in the early days of CCTM based AI research, there was also another allegedly distinct approach, one based upon network models. The work on network models seems to fall broadly within the scope of the term 'connectionist' (see Aizawa 1992), although the term had yet to be coined at the time. These two approaches were "two daughter sciences" according to Papert (1988: p. 3). The fundamental difference between these two 'daughters', lay (according to Dreyfus and Dreyfus (1988: p. 16)) in what they took to be the paradigm of intelligence. Whereas the early connectionists took learning to be fundamental, the traditional school concentrated upon problem solving.

Although research on network models initially flourished along side research inspired by the CCTM, network research fell into a rapid decline in the late 1960's. Minsky (aided and abetted by Papert) is often credited with having personally precipitated the demise of research in network models, which marked the end of the first phase of connectionist research. Hecht-Nielson (1990: pp. 16-17) describes the situation (as it is presented in standard versions of the early history of connectionism) thus,

> The final episode of this era was a campaign led by Marvin Minsky and Seymour Papert to discredit neural network research and divert neural network research funding to the field of "artificial intelligence"....The campaign was waged by means of personal persuasion by Minsky and Papert and their allies, as well as by limited circulation of an unpublished technical manuscript (which was later de-venomized and, after further refinement and expansion, published in 1969 by Minsky and Papert as the book Perceptrons).[1]

In Perceptrons, Minsky and Papert (1969) argued that there were a number of fundamental problems with the network research program. For example they argued that there were certain tasks, such as the calculation of topological function of connectedness and the calculation of parity, which Rosenblatt's perceptrons[2] could not solve. The inability to calculate parity proved to be particularly significant, as this showed that a perceptron could not learn to evaluate the logical function of exclusive-or (XOR). The results of Minsky and Papert's (1969: p. 231-232) analysis lead them to the conclusion that, despite the fact that perceptrons were "interesting" to study, ultimately perceptrons and their possible extensions were a "sterile" direction of research.

The publication of Perceptrons was not the only factor in the decline of network research in the late Sixties and early Seventies, though. A number of apparently significant research successes from the non-network approach, also proved to be influential. Systems such as Bobrow's (1969) STUDENT, Evan's (1969) Analogy program and Quillian's (1969) semantic memory program called the Teachable Language Comprehender, were demonstrated. These systems, which had properties like those associated with the CCTM, did not appear to suffer from the limitations that afflicted network models.[3] Indeed, these systems seemed to show considerable promise with respect to emulating aspects of human cognition. Bobrow's STUDENT program, for example, was designed to solve algebra word problems. In doing this, the program would accept input in (a restricted subset of) English. This property of the system lead Minsky (1966: p. 257) to claim that "STUDENT...understands English". Although this is now seen to be highly misleading (see, for example Dreyfus' 1993: pp. 130-145 critiques of all the systems mentioned above), at the time it was a fairly impressive claim which did broadly seem to be supported by Bobrow's program. Network research, by comparison, had nothing as impressive to offer. Given Minsky and Papert's unfavorable conclusions and the apparent fruitfulness of non-network based approaches, it is not surprising that research into network systems went into decline.

---

[1] Some of the hostility described in this account is confirmed by Papert (1988: pp.4-5).

[2] Perceptron based systems were, arguably, the flag-ship variety of network systems at the time.

During the 1970s, there was very little work done on connectionist style systems. Almost all the research done in AI concentrated upon the other approach. This is not to say that there was no network research done during this period. A few individuals, most notably Anderson (1972), Kohonen (1972) and Grossberg (1976), did continue to investigate connectionist systems, however network researchers were very much the exception rather than the rule. After a ten year hiatus though, connectionism reappeared on the scene as a significant force. One reason for this resurrection was that a number of technical developments were made which seemed to indicate that Minsky and Papert had been premature to write off such systems .

---

[3] It is worth noting that all the systems mentioned here were developed by Minsky's own graduate students, according to Dreyfus (1993: p. 149). For a more detailed overview of each of these programs and the way they were evaluated, see Dreyfus (1993: pp. 130-145).

Minsky and Papert only considered Rosenblatt's perceptrons in their book of the same name. One of the significant limitations to the network technology of the time was that learning rules had only been developed for networks which consisted of two layers of processing units (i.e. input and output layers), with one set of connections between the two layers. However, Minsky and Papert (1969: p. 232) had conjectured (based on what they termed an "intuitive" judgment) that extensions of the perceptron architecture, for example based upon additional layers of units and connections, would be subject to limitations similar to those suffered by one-layer perceptrons. By the early 1980s more powerful learning rules had been developed which enabled multiple-layered networks to be trained. The results that such multiple-layered networks yielded indicated that Minsky and Papert's 'intuitive judgment' was too hasty (see Rumelhart and McClelland 1987: pp. 110-113).[4]

Another important factor in the renaissance of network models, according to the standard view, was a growing dissatisfaction with the traditional approach. Arguably the most important event in this renaissance was the publication of the two volume work Parallel Distributed Processing by Rumelhart, McClelland et al. (1987).[5] Dreyfus and Dreyfus (1988: pp. 34-35) describe the situation thus,

---

[4] For a more detailed account of the work which underwrote the rebirth of connectionism, as well as a more detailed account of network research during the 1970s and early 1980s, see McClelland, Rumelhart and Hinton (1987: pp. 41-44).

[5] It is now standard practice to refer to this work by the title The PDP Volumes.

> Frustrated AI researchers, tired of clinging to a research program that Jerry Lettvin characterized in the early 1980s as "the only straw afloat," flocked to the new paradigm [*sic*]. Rumelhart and McClelland's book...sold six thousand copies the day it went onto the market, and thirty thousand are now in print.

Smolensky (1988) describes how "...recent meetings [i.e. those circa 1988] of the Cognitive Science Society have begun to look like connectionist pep rallies.". Hecht-Nielsen explicitly (1990: p. 19) describes those who came 'flocking' to the new connectionism as 'converts'. The religious analogy is not insignificant here. Just as it is often the case that religious converts seek to vilify other belief systems, so the converts to connectionism often attempted to emphasize what they believed to be the fundamental differences between the connectionist and the CCTM based approach. Of course, such an environment is highly conducive to the development of myths. (This may at least partially account for the existence of the myths of connectionism, discussed in Chapter IV).

So, the history of connectionism as commonly characterized, is a history which, apart from the early years, has been marked by a struggle with the approach which had roots in the assumptions underlying the CCTM. Many recent descriptions of the relationship between the approaches dwell almost exclusively upon the putative differences between them. For example, Schneider (1987), Churchland (1989), Smolensky (1991), Sterelny (1990), Cummins (1991), Tienson (1991), Bechtel and Abrahamsen (1991), Fodor and Pylyshyn (1988) and Hecht-Nielsen (1991) all portray the two approaches as being in direct competition with one another. Given the standardly told story of the history of connectionism, such an antagonistic relationship between the two approaches is far from surprising. The standard version of this history also suggest that certain episodes (such as the publication and circulation of Perceptrons) were marked by a certain guile and personal crusading on the part of the anti-connectionist camp. Connectionism is usually portrayed as a field of research which was unfairly retarded early on, but which, due to the publication of The PDP Volumes and the empirical inadequacies of the alternative, has only comparatively recently begun to bloom. This kind of perspective fits well with the view that connectionism provides the basis of some kind of substantial alternative to the assumptions underlying the CCTM. Unfortunately, this version of history is highly selective, partial and in certain respects, down right misleading.

As a matter of historical fact, in the early days of AI research, a number of high profile researchers in the field worked with *both* approaches. Even Papert (1988: p. 10) for example, did work on network models. Another example is von Neumann, who worked with McCulloch-Pitts nets and showed that such nets could be made reliable and (moderately) resistant to damage by introducing redundancy (i.e. having several units do the job of one). In fact, von Neumann published quite extensively on the topic of networks (see von Neumann 1951, 1956 and 1966), although his name is most often associated with classical systems.

There were a number of significant results which came to light in the 1940's and 1950's, with respect to network models. Arguably the most important of these was McCulloch and Pitt's (1943) demonstration that networks of simple interconnected binary units (which they called 'formal neurons'), when supplemented by indefinitely large memory stores, were computationally equivalent to a Universal Turing Machine. Later, Rosenblatt

(1958) developed an improved version of the units employed by McCulloch and Pitts. Both McCulloch and Pitt's formal neurons and Rosenblatt's units had threshold activation functions (by contrast, most modern connectionist units have continuous activations).[6] The innovation which Rosenblatt made was to develop modifiable continuously valued connections (i.e. weights) between the units. This enabled networks of these units to be effectively trained. In particular, Rosenblatt's training procedure was supervised and such that the system learned only when it made a 'mistake' with respect to the desired output for a particular input pattern. Rosenblatt called networks of his units 'Perceptrons'.

The significance of Rosenblatt's innovation became clear when he (1962) demonstrated the Perceptron Convergence Theorem. This theorem holds that if there is a set of weighted connections of a perceptron, such that the perceptron gives the desired responses for a set of stimulus patterns, then after a finite number of presentations of the stimulus-response pairs and applications of the training procedure, the perceptron will converge upon that set of weights which would enable it to respond correctly to each stimulus in the set.[7]

Marvin Minsky, so often portrayed as a villain in the standard version of the history of connectionism, has also made significant contributions to network research. In 1951 Minsky, in conjunction with Dean Edmonds, constructed a machine known as the SNARC (Rumelhart and Zipster (1987: pp. 152-154)). The SNARC was the first 'learning' machine and was constructed along what would now be thought of as connectionist principles, according to Hecht-Nielson (1990: p. 15). Indeed, his work with the SNARC formed the basis of Minsky's Ph.D. dissertation. Minsky (1954) even included the phrase 'neural nets' in the title of his dissertation. According to Minsky (personal communication, 1994) it wasn't until "...around 1955, largely at the suggestion of my friend Ray Solomonoff....[that] I moved toward the direction of heuristic serial problem solving.". That is to say, Minsky's interest in network based system in fact predates his interest in CCTM based systems.

---

[6] See the discussion of activation functions in Chapter III, for further details.

[7] For a more detailed account of the history of network models, see Cowan and Sharp (1988).

It is also the case that in the early phase of connectionist research, there was relatively little antagonism between the two approaches. The difference was rather one of attitude. Minsky (personal communication, 1994) characterizes the situation as follows,

> ...Nilsson [a network researcher from Stanford] was a good mathematician, as were we, so this attitudinal split had no important effect on what both sets of pioneers actually did; both groups did in fact try to understand why each method worked on some problems but not on others.

These facts are perhaps somewhat surprising, given the malevolent role ascribed to Minsky in the standard histories of connectionism. Perhaps, it might be conjectured, the adversarial relationship between the approaches derives from Minsky and Papert's critique of networks in Perceptrons. If this is the case for some though, this adversarial perspective does not seem to be shared by Minsky himself. Even long after the publication of Perceptrons, Minsky continued to do theoretical work upon network models. In 1972 for example, Minsky (1972: p. 55) published a proof that showed that "Every finite state machine is equivalent to, and can be 'simulated' by, some neural net". Indeed, Minsky does not endorse the adversarial view of the relation between the approaches even today. Consider the following remark by Minsky (1990),

> Why is there so much excitement about Neural Networks today, and how is this related to research on Artificial Intelligence? Much has been said, in the popular press as though these were conflicting activities. This seems exceedingly strange to me, because both are parts of the same enterprise.

These facts serve to show that the supposed distinction between the two approaches, at least in the early days of network research, were not as sharp as some commentators would have us believe (C.f. Dreyfus and Dreyfus (1988)). Furthermore, there seem to be grounds for wondering just who is responsible for the putative conflict between the approaches. Although he is frequently 'demonised' in the connectionist literature, it does not seem to be Minsky!

The responsibility for the antagonistic relation between the approaches, and the consequently partial standard history, does not straightforwardly lie with any one individual or group. It is rather the consequence of a number of factors. It is certainly the case that the authors of the PDP Volumes must take some of the responsibility. For example, McClelland, Rumelhart and Hinton (1987: p. 11) remark that

> PDP models...hold out the hope of offering computationally sufficient and psychologically accurate mechanistic accounts of the phenomena of human cognition which have eluded successful explication in conventional computational formalisms...

Such remarks are fairly clearly antagonistic to advocates of the more traditional approach. There are many other similar examples which can be found in the PDP Volumes.

It is also the case that the authors of the PDP Volumes make a number of claims about the relationship between their systems and the ones discussed by Minsky and Papert in Perceptrons which are not entirely accurate. Examples of misleading claims can be found in Rumelhart, Hinton and McClelland (1986: p. 65), Rumelhart and McClelland (1986: p. 113) and Rumelhart, Hinton and Williams (1986: p. 361), for example. Minsky and Papert's responses to these specific claims are in the epilogue of the third edition of Perceptrons (1988). Of course, the authors of the PDP Volumes were not alone in misunderstanding Minsky and Papert's work. Minsky (personal communication, 1994) describes the situation thus,

> It would seem that Perceptrons has much the same role as The Necronomicon -- that is, often cited but never read.

It is by no means the case though that the responsibility for the adversarial relationship between connectionism and approaches which share assumptions with the CCTM belongs just to the authors of the PDP Volumes. In fact, Rumelhart (personal communication, 1994) still considers his work as part of the more general enterprise of AI. He also believes that the 'AI is dead' talk which arose just after the publication of the PDP Volumes, was mistaken. Undoubtedly, the emergence of 'new' connectionism was accompanied by a certain amount of jumping on the proverbial connectionist bandwagon. It is almost certainly the case that a number of the new 'converts' to connectionism made claims which were far too strong and thereby engendered the wrath of some of the advocates of the other approach. This too is likely to have encouraged an antagonistic relation between the two approaches. It is also certainly the case that some of the antagonism between the approaches can be traced backed to Fodor and Pylyshyn's (1988) paper.

Although it would be possible to pursue this theme in much greater detail, I hope that the above is sufficient to make it clear that this putative antagonism between CCTM and connectionist approaches to studying the mind is, for the most part, a comparatively recent phenomenon. It is interesting and (I believe) significant to note that some of the major figures in the fields (e.g. Rumelhart and Minsky) do not subscribe to this view of the relationship.

The standardly told historical story clearly encourages the view that connectionism is an alternative to the CCTM. This view, in conjunction with the superficial structural differences between connectionist networks and devices such as Turing machines, helps in part to explain why connectionist systems might seem an apparently plausible basis for an alternative conception of a computational theory of mind. This plausibility notwithstanding though, the facts of the matter, as described and discussed in the previous chapters, show that such a conclusion does not follow. Rather, connectionist research serves to open upon a host of possibilities with respect to refining the key notions of the CCTM and related positions. In the next section, I will very briefly sketch what I take to be some of the potentially most fruitful lines of further inquiry.

Further Research Directions: The Future

The results described in the previous chapters open up a number of possible future avenues of research. As these avenues divide quite naturally between research in cognitive science and research in philosophy, it is perspicuous to describe them separately. However, it is important to realize that they do together form part of a unified (though interdisciplinary) research strategy, as the results and conclusions from each area will importantly influence the work in the other.

Cognitive Science

The potential of the analytic methodology described by Berkeley *et al.* (1995) has not been fully explored. Given the promising nature of the results gained from analyzing the network L10 though, it would be interesting and useful to see this methodology applied to a range of other problems. Although Berkeley *et al.* (1995) do discuss the deployment of their method in various domains (ranging from a variation of Hinton's 1986 kinship problem, to a 6-bit parity problem), there are still a large number of potentially interesting problems which have yet to be tackled, for example Seidenberg and McClelland's (1989) word recognition and naming problem, upon which McCloskey based his critique of the connectionist research methodology. In addition, it might be particularly interesting to investigate the structures developed within networks trained upon a range of logic problems, more extensive than the set described by Bechtel and Abrahamsen (1991), in order to see whether such systems developed rules analogous to those described for the network L10.

However, it is unlikely that a single analytic technique will suffice for all classes of problems or network types. For this reason, the work described in the previous chapters lends further credibility to Clark's (1993) call for the development of a variety of analytic and interpretative methodologies which can be used to understand the functioning of trained connectionist networks. Although there has been some pessimism expressed in the literature about the possibility of understanding trained networks, the evidence discussed here shows that this pessimism may be unwarranted. Furthermore, because of the importance of network interpretation to the viability of the connectionist research program in cognitive science, further development in this direction is crucial.

Perhaps the most strongly indicated future direction for research though, is that towards the development of connectionist models which exhibit higher degrees of systematicity (in Hadley's 1994 sense). Even though the network described by Berkeley *et al.* (1995) was only weakly systematic, it served to provide results which were highly illuminating. In particular, it provided grounds for proposing that the notion of 'compositionality' (as understood by Fodor and Pylyshyn 1988) may come in degrees in a manner analogous to the way that systematicity does. However, in order to do this, several steps need to be taken. Probably the most important of these would be the development of a connectionist model, or the extension of a currently existing model, so that embedding could be handled. This alone would not be sufficient though. It would also be necessary to develop training and test sets of data for such models that are sufficiently large and appropriately structured so as to be able to meet all of Hadley's (1994) conditions for the higher degrees of systematicity. Steps are already being taken in this direction. For example, Hadley and Hayward (1997) have recently developed a connectionist style model, based upon Hebbian learning, which can learn to exhibit strong systematicity. However, Hadley and Haywards (1997) model is only a beginning. There are many other connectionist architectures which need to be explored. Furthermore, some of the representational assumptions made by Hadley and Hayward in their model are such that it does not enable them to claim to have a full counter-example to Fodor and Pylyshyn's (1988) thesis.

There is also plenty of potential for the development of connectionist systems, which may be able to exhibit compositionality of the higher kinds. For example, some time ago I developed a representational format known as 'Connectionist Polish Notation' (CPN), which enables connectionist systems to represent formulae of sentential logic with embeddings of arbitrary degree.[8] A natural extension of the work on L10 would be to try to develop and analyze a system which employed CPN, in the hope of producing a network which could exhibit at least quasi-compositionality.

A final future direction for further investigation within the strictly cognitive domain would be to study the performance of networks on various problems, in comparison to the performance of biological cognitive agents. Only if a plausible case could be made that a network was strongly equivalent (in Pylyshyn's 1984 sense) to biological cognitive agents will their be any basis upon which inferences could be drawn about human cognitive functioning. This would be an important step in moving the argument about connectionist systems on from merely showing that they could be *in principle* models of cognitive functioning, to them *actually being* such models. It is only under this condition that the full explanatory potential of trained network models could be evaluated. It may turn out to be the case, for example, that it would only be possible to develop models which were strongly equivalent, by integrating elements discovered in network models into more traditional architectures. Nonetheless, further research on connectionist models which can handle embedded structure will be necessary to determine the extent to which this might be the case.

---

[8] At the present time, this work is unpublished.

Although such technical developments in cognitive science would be very helpful, they alone would not serve to answer all of the questions raised by the analysis of the logic network, with respect to the relationship between connectionist systems and the CCTM. There is also a considerable amount of strictly philosophical work which will also be required.

Philosophy

From the discussion in the previous chapter, the notion of 'tokenhood' is a prime candidate for further philosophical consideration. In particular, it is unclear how the potential complexity of embedding which can be supported relates to the status of an entity as a token. Indeed, the consideration of token-like items within connectionist networks may throw light upon hitherto unexpected degrees of freedom within the idea of something being a 'token'. Related notions, such as that of 'symbolhood' might also turn out to require additional clarification. However, the kinds of data which such a conceptual analysis would have to pay attention to would have to extend beyond the standardly considered examples from language and logic. If the avenues of cognitive science research discussed above were to be pursued successfully, a whole new class (or even classes) of data might have to be factored into such reflections. In addition, any new evidence about novel forms of compositionality in connectionist systems (should any turn up) will be relevant to the conceptual analysis of the notion of tokenhood too.

Similarly, the whole idea of what it is for something to be a 'rule' or an 'operation' may have to be extended in the light of 'rules' discovered within networks. For example, the rules discovered in the network L10 had many of the properties usually associated with traditional rules of deductive inference. However, they were discovered by the network as a result of a process which is fundamentally inductive in nature. This may make it appear initially as if deduction is in fact based in induction. Such a conclusion would be no surprise to those such as Dreyfus and Dreyfus (1986) who have argued, along with Ryle (1949), that skills (knowing how to do something) are not reducible to declarative knowledge (knowing that certain things are the case). However, the fact that the rules discovered in the network are not (obviously) as widely applicable as those of natural deduction, would suggest caution, initially at least. Once again, the strength of the conclusions which could be drawn about rules and operations would depend upon the results of the further empirical work suggested above.

Hadley (1994) has already begun some of the conceptual work necessary in trying to get the idea of systematicity clear, such that it can be usefully employed in the context of connectionist systems. However, there are grounds for believing that this work could be extended further. The fact that Hadley's three degrees of systematicity all incorporate an element of learning means that there may be cases in which it is unclear how they map onto systems which do not explicitly undergo training.[9] Although it might be argued that the programming of a Turing machine is learning is some sense, it is far from clear whether the relationship between learning in the context of an automated learning device, such as a network, and learning in this sense are straightforwardly related to one another. In addition, it has yet to be shown that either of these two types of learning are have relevant analogues in the case of biological cognitive agents.

---

[9] It has recently come to my attention that Hadley (1996) has addressed the issue of systematicity in non-learning systems.

A related problem, which also may have potential impact on Hadley's (1994) views on systematicity, is the lack of a widely applicable metric for generalization. This problem seems to relate to the difficulties already raised about the nature of rules and operations and their relation to traditional philosophical difficulties surrounding induction. Perhaps, with new evidence from connectionist systems in hand, some philosophical headway may be made on these problems.

The results of analyzing connectionist networks may provide philosophical insights into what Cummins (1989) has called the 'problem of representations' (plural). This is the problem of determining the kinds of entities which can function as mental representations, and the properties of those entities. This in turn may have an impact on what Cummins calls the 'problem of representation' (singular). The problem of representation (singular) is the problem of determining what it is for one thing to be a representation of another. The two problems are closely related to one another because (Cummins 1989: pp. 1-2) the solution accepted to the problem of representations (plural) acts as a constraint upon the kinds of accounts which are acceptable to the problem of representation (singular). For example, Cummins (1989: pp. 84-86) feels compelled to reject Millikan's (1984 and 1991) proposed solution to the problem of representation (singular) on the grounds that her solution depends crucially upon a systems having a history of the correct kind. His solution to the problem of representations (plural) is the CCTM, which he believes is fundamentally ahistorical in nature. Given the standard course of development of connectionist networks (including the pilot tests run by experimenters, which are seldom reported in the literature), it might be possible to make a case that networks have a history of the appropriate kind so that Milikan's account of representation (singular) could be made to apply to them. This too would be an interesting line of philosophical research which might be pursued.

Finally, many of the concepts mentioned above, which seem to be in need of further philosophical treatment, have roles to play not only in cognitive science and the philosophy of mind, but also in other areas of philosophy, such as the philosophy of language and the philosophy of logic. This being the case, there may well be all sorts of significant philosophical insights to be gained by applying the lessons learned from considering the implication of network research for these notions in other domains. The extent to which these insights prove helpful though, will depend upon the results of the further investigation of the properties of connectionist systems and the philosophical conclusions drawn from these results. In addition, should it turn out that these results show that the connectionist approach is not much different from the CCTM, then a number of the arguments which have been proposed against the CCTM (e.g. by Dreyfus 1991 or by Searle 1980 and 1992) might also turn out to apply to connectionism.[10]

In concluding, a remark made by Pylyshyn (1984: p. 69) comes to mind. He observed that,

> ...despite some 50 years of study (starting from Turing's famous paper on computability), there is still no consensus on just what are the essential elements of computing.

---

[10] This might lead to the consequence that, contrary to current thought, what Searle and Dreyfus are really attacking is the one thing which the CCTM and connectionism already agree upon, the representational theory of mind (see Sterelny 1991).

Determining the 'essential elements of computing' sounds like a prototypical example of the kind of ontological question which has been the domain of philosophers for millennia. Thus, this would seem to be a job not only for computer scientists, but also for philosophers. Given the centrality that the notion of computation has come to play in philosophical theorizing, especially about the mind, it can be little short of shocking that no progress has been made on this topic. Indeed, it is only recently that such issues have become subject to detailed and extensive philosophical scrutiny (see for example Hardcastle 1996 and Smith 1996). However, one possible explanation for this state of affairs is that, by and large until recently, central notions of computation have only been considered within the domain of the CCTM. When it comes to questions like 'what is a token?', 'what is an operation?', 'what are systematicity and compositionality?', it is only the emergence of connectionist networks which has brought the difficulties inherent with these notions to the fore. As Wittgenstein (1953: 593) famously observed,

> A main cause of philosophical disease--a one-sided diet: one nourishes one's thinking with only one kind of example.

Considering the results from research into connectionist systems may well serve to offer philosophers just the 'dietary supplement' they need to come to grips with such questions and perhaps provide, at long last, substantial answers.

Bibliography

Adams, F., Aizawa, K. and Fuller, G. (1992), "Rules in Programming Languages and Networks" in Dinsmore (1992: pp. 49-67).

Aizawa, K. (1992), "Connectionism and Artificial Intelligence: History and Philosophical Interpretation", in The Journal of Experimental and Theoretical Artificial Intelligence, 4, pp. 295-313.

Aizawa, K. (1994), "Representations without Rules, Connectionism and the Syntactic Argument", in Synthese, 101/3, pp. 465-492.

Anderson, J. (1972), "A simple neural network generating an interactive memory", in Mathematical Biosciences, 14, pp. 197-220.

Audi, R. (Ed.), (1995), The Cambridge Dictionary of Philosophy, Cambridge U. P. (Cambridge).

Ballard, D. (1986), "Cortical structures and parallel processing: structure and function", in The Behavioural and Brain Sciences, 9, pp. 67-120.

Barnden, J. and Pollack, J. (Eds.), (1991), Advances in Connectionist and Neural Computation Theory (Vol. 1): High-Level Connectionist Models, Ablex Pub. Co. (Northwood, NJ).

Bates, E. (1996), Discussion of Ling (1996). See Cottrell (1996: pp. 61-68).

Baumgartner, P. and Payr, S. (Eds.), (1995), Speaking Minds: Interviews with Twenty Eminent Cognitive Scientists, Princeton U.P. (Princeton, NJ).

Bechtel, W. (1985), "Contemporary Connectionism: Are The New Parallel Distributed Processing Models Cognitive Or Associationist?" in Behaviourism, 13/1, pp. 53-61.

Bechtel, W. (1994), "Natural Deduction in Connectionist Systems" in Synthese, 101, pp. 433-463.

Bechtel, W. and Abrahamsen, A. (1991) Connectionism and the Mind, Basil Blackwell (Cambridge, Mass.).

Bergmann, M., Moor, J., & Nelson, J. (1990), The Logic Book McGraw-Hill (New York).

Berkeley, I., Dawson, M., Medler, D., Schopflocher, D., and Hornsby, L. (1995), "Density Plots of Hidden Unit Activations Reveal Interpretable Bands", in Connection Science, Vol. 7, No. 2, pp. 167-186.

Berkeley, I. S. N., (1996), "Connectionism, Tri-Level Functionalism and Causal Roles", in Two Sciences of The Mind: Readings in Cognitive Science and Consciousness, Ed. S. O'Nuallain, P. McKevitt and E. Mac Aogain, John Benjamins (Amsterdam), pp.219-231 (forthcoming).

Best, J. (1986), Cognitive Psychology, West Pub. Co. (New York).

Block, N. (1978), "Troubles with Functionalism", in Block (1980, Vol. 1: pp. 268-305).

Block, N. (1980a), "Introduction: What Is Functionalism?" in Block (1980: pp. 171-184).

Block, N. (Ed.), (1980), Readings in Philosophy of Psychology, (2 Vols.), Harvard U. P. (Cambridge, Mass.).

Bobrow, D. (1969), "Natural Language Input for a Computer Problem Solving Program", see Minsky (1969).

Boden, M. (1981), Minds and Mechanisms: Philosophical Psychology and Computational Models, Harvester Press (Brighton).

Boden, M. (1990), (Ed.), The Philosophy of Artificial Intelligence, Oxford U. P. (Oxford).

Cederblom, J. and Paulsen, D. (1991), Critical Reasoning (3rd Ed.), Wadsworth Pub. Co. (Belmont, CA).

Chalmers, D. (1990), "Why Fodor and Pylyshyn Were Wrong: the Simplest Refutation", in Proceedings of the Twelfth Annual Conference of the Cognitive Science Society, (Cambridge, Mass.).

Chambers, J., Cleveland, W., Kleiner, B. and Tukey, P., (1983), Graphical Methods of Data Analysis, Wadsworth (Belmont, CA).

Churchland, P. M. (1989), The Neurocomputational Perspective: The Nature of Mind and the Structure of Science, MIT Press (Cambridge, Mass.).

Churchland, P. M. (1990), Matter and Consciousness: A Contemporary Introduction to the Philosophy of Mind, MIT Press (Cambridge, Mass.).

Churchland, P. S. and Sejnowski, T. (1994), The Computational Brain, MIT Press (Cambridge, Mass.).

Clark, A. (1989), Microcognition: Philosophy, Cognitive Science and Parallel Distributed Processing, MIT Press (Cambridge, Mass.).

Clark, A. (1993), Associative Engines: Connectionism, Concepts, and Representational Change, MIT Press (Cambridge, Mass.).

Clark, A. and Lutz, R. (Eds.), (1992), Connectionism in Context, Springer-Verlag (Heidelberg).

Clark, A. and Lutz, R. (1992a), “Introduction” in Clark and Lutz (1992: pp. 1-15).

Clark, A. and Thronton, C. (1996), “Trading Spaces: Computation, representation, and the limits of uninformed learning”, in Behavioral and Brain Sciences, (forthcoming).

Cottrell, G. (Ed.), (1996), Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society, (held in La Jolla, CA, 12-15 July 1996), LEA (Mahwah, NJ).

Cowan, J. and Sharp, D. (1988), "Neural Nets and Artificial Intelligence", in Graubard (1988: pp. 85-121).

Cummins, R. (1989), Meaning and Mental Representation, MIT Press (Cambridge, Mass.).

Davidson, D. (1978), "What Metaphors Mean", reprinted in Martinich (1985: pp. 438-448).

Dawson, M. and Berkeley, I. (1993), "Making a Middling Mousetrap", commentary on Shastri, L. and Ajjanagadde, V. (1993) ""From Simple Associations to Systematic Reasoning", in Behavior and Brain Sciences, Vol. 16, No. 3, pp. 454-455.

Dawson, M., Medler, D., and Berkeley, I. (1997), "PDP Networks Can Provide Models That Are Not Mere Implementations of Classical Theories", in Philosophical Psychology, (forthcoming).

Dawson, M. and Schopflocher, D. (1992), "Modifying the Generalized Delta Rule to Train Networks of Non-monotonic Processors for Pattern Classification", in Connection Science, 4/1, pp. 19-31.

Dawson, M. and Schopflocher, D. (1992a) "Autonomous Processing in Parallel Distributed Processing Networks", in Philosophical Pschology, Vol. 5, No. 2, pp. 199-219.

Dawson, M. and Shamanski, K. (1993), "Connectionism, Confusion, and Cognitive Science", in Journal of Intelligent Systems, in press.

deGroot, J. and Chusid, J. (1988), Correlative Neuroanatomy, (12th Ed), Appleton and Lange (Connecticut).

Dennett, D. (1991), Consciousness Explained, Little, Brown and Co. (Boston).

Devitt, M. (1990), "A Narrow Representational Theory of Mind" in Lycan (1990: pp. 371-398).

Dinsmore, J. (Ed.), (1992) The Symbolic and Connectionist Paradigms: Closing The Gap Lawrence Erlbaum Associates (Hillsdale, New Jersey).

Dretske, F. (1981), Knowledge and the Flow of Information, MIT Press (Cambridge, Mass.).

Dretske, F. (1988), Explaining Behavior: Reasons in a World of Causes. MIT Press (Cambridge, Mass.).

Dreyfus, H. (1993), What Computers *Still* Can't Do, MIT Press (Cambridge, Mass.).

Dreyfus, H. and Dreyfus, S. (1986), Mind over Machine: The power of human intuition and expertise in the era of the computer, The Free Press (New York).

Dreyfus, H. and Dreyfus, S. (1988), "Making a Mind Versus Modeling the Brain: Artificial Intelligence Back at a Branchpoint", in Graubard (1988: pp. 15-43).

Egan, F. (1995), "Computation and Content" in The Philosophical Review, 104/2, pp. 181-203.

Elman, J. (1990), "Finding Structure in Time", Cognitive Science, 14, pp. 179-212.

Evans, T. (1969), "A Program for the Solution of a Class of Geometric-Analogy Intelligence Test Questions", see Minsky (1969).

Feigenbaum, E. and Feldman, J. (Eds.), (1963), Computers and Thought, McGraw-Hill (New York).

Feldman, J. (1985), "Connectionist Models and their applications: Introduction", in Cognitive Science 9, pp. 1-2.

Feldman, J. and Ballard, D. (1982), "Connectionist Models and Their Properties", in Cognitive Science 6, pp. 205-254.

Field, H. (1978), "Mental Representation", in Erkenntnis 13, pp. 9-61.

Flew, A. (Ed.), (1951), Essays on Logic and Language, Blackwells (Oxford).

Fodor, J. (1975), The Language of Thought, Harvard U. P. (Cambridge, Mass.).

Fodor, J. (1980), "Methodological Solopsism considered as a research strategy in cognitive psychology", in Behavioral and Brain Sciences, 3, pp. 63-73.

Fodor, J. (1987), Psychosemantics, MIT Press (Cambridge, Mass).

Fodor, J. and Pylyshyn, Z. (1988), "Connectionism and Cognitive Architecture: A Critical Analysis", in Cognition 28, pp. 3-71.

Forbus, K. and de Kleer, J. (1992), Building Problem Solvers, MIT Press (Cambridge, Mass.).

Getting, P. (1989), "Emerging Principles Governing the Operation of Neural Networks", in Annual Review of Neuroscience, 12, pp. 184-204.

Glymour, C. (1980), Theory and Evidence, Princeton U. P. (Princeton, NJ).

Goschke, T. and Koppelberg, D. (1991), "The Concept of Representation and the Representation of Concepts in Connectionist Models", in Ramsey, Stich and Rumelhart (1991: pp. 129-161).

Govier, T. (1992), A Practical Study of Argument, (3rd Ed.), Wadsworth Pub. Co. (Belmont, CA).

Graubard, S. (Ed.), (1988), The Artificial Intelligence Debate: False Starts, Real Foundations, MIT Press (Cambridge, Mass.).

Grossberg, S. (1976), "Adaptive pattern classification and universal recoding: I. Parallel development and coding in neural feature detectors", in Biological Cybernetics, 23, pp.121-34.

Hadley, R. (1994), “Systematicity in Connectionist Language Learning” in Minds and Machines, 9, pp. 247-272. (N.B. All page references in the text are to a manuscript copy.)

Hadley, R. (1996) "Cognition, Systematicity, and Nomic Necessity", in Mind and Language, (in press).

Hadley, R. and Hayward, M. (1997), “Strong Semantic Systematicity from Hebbian Connectionist Learning”, to appear in Minds and Machines. (N.B. All references to this paper are to a manuscript version).

Hardcastle, V. (1996), "Computationalism", in Synthese, 105, pp. 303-317.

Haugeland, J. (1985), Artificial Intelligence: The Very Idea, MIT Press (Cambridge, Mass.).

Hebb, D. (1949), The Organisation of Behaviour, John Wiley (New York).

Hecht-Nielsen, R. (1990), Neurocomputation, Addison-Wesley Pub. Co. (New York).

Hodges, A. (1983), Alan Turing: The Enigma, Burnett Books (London).

Honavar, V. and Uhr, L. (Eds.), (1993), Symbol Processing and Connectionist Models in Artificial Intelligence and Cognition: Steps Towards Integration, Academic Press (New York).

Hopcroft, J. and Ullman, J., (1969), Formal Languages and Their Relation to Automata, Addison-Wesley Pub. Co. (London).

Hopcroft, J. and Ullman, J. (1979), Introduction to Automata Theory, Languages, and Computation, Addison-Wesley Pub. Co. (London).

Horgan, T. and Tienson, J. (1990), “Representations without rules”, in Philosophical Topics, 17(1), pp. 147-174.

Horgan, T. and Tienson, J. (1991), Connectionism and The Philosophy of Mind, Kluwer Academic (Dordrecht).

Jacobowitz, H. (1963), Electronic Computers, Doubleday & Co. (New York).

Jeffress, L. (Ed.), (1951), Cerebral Mechanisms in Behaviour, Wiley (New York).

Johnson-Laird, P. (1983), Mental Models Harvard U. P. (Cambridge, Mass.).

Johnson-Laird, P., and Byrne, R. (1991), Deduction, Lawrence Erlbaum Associates (Hillsdale).

Kithcher, P. (1993), The Advancement of Science, Oxford U. P. (Oxford).

Kohonen, T. (1972), "Correlation matrix memories", IEEE Transactions on Computers, C-21, pp. 353-9.

Kolb, B. and Whishaw, I. (1990), Fundamentals of Human Neuropsychology (3rd Ed.), Freeman and Co. (New York).

Lachter, J. and Bever, T. (1988), "The relationship between linguistic structure and associative theories of language learning: A constructive critique of some connectionist learning models", in Pinker and Mehler (1988: pp. 195-247).

Ling, C. (1996), "Can Symbolic Algorithms Model Cognitive Development?" (Abstract) in Cottrell (1996: pp. 67-68).

Lipsey, R. (1979), Positive Economics, (5th Ed.), Weidenfield and Nicolson (London).

Lloyd, D. (1989), Simple Minds, MIT Press (Cambridge, Mass.).

Loewer, B. and Rey, G. (Eds.), (1991) Meaning In Mind: Fodor and his Critics, Blackwell (Oxford).

Lycan, W. (1990), Mind and Cognition: A Reader, Basil Blackwell (Oxford).

MacDonald, C. and MacDonald, G. (Eds.), (1994), The Philosophy of Psychology: Debates on Psychological Explanation, Basil Blackwell (Oxford).

Martin, R. (1991), The Philosopher's Dictionary, Broadview Press (Scarborough, Ont.).

Martinich, A. (1985), (Ed.), The Philosophy of Language, Oxford U. P. (Oxford).

McClelland, J. and Rummelhart, D. (1988), Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises, MIT Press (Cambridge, Mass.).

McClelland, J., Rumelhart, D. and Hinton, G. (1987), "The Appeal of Parallel Distributed Processing", in Rumelhart, McClelland, et al. (1987: Vol. 1, pp. 3-44).

McCloskey, M. (1991), "Networks and Theories: The Place of Connectionism in Cognitive Science" in Psychological Science, Vol. 2, No. 6, pp. 387-395.

McCulloch, W. and Pitts, W. (1943), "A Logical Calculus of the Ideas Immanent in Nervous Activity", in Bulletin of Mathematical Biophysics, 5/115.

Meyering, T. (1989), Historical Roots of Cognitive Science: The Rise of a Cognitive Theory of Perception from Antiquity to the Nineteenth Century, Kluwer Academic (Dordrecht).

Miller, G., Galanter, E. and Pribram, K. (1960), Plans and the Structure of Behavior Holt (New York).

Millikan, R. (1984) Language, Thought and Other Biological Categories, MIT Press (Cambridge, Mass.).

Millikan, R. (1991), White Queen Psychology and Other Essays for Alice, MIT Press (Cambridge, Mass.).

Minsky, M. (1954), "Neural nets and the brain-model problem", Unpublished Doctoral Dissertation, Princeton University.

Minsky, M. (1966), "Artificial Intelligence" in Scientific American 215/3.

Minsky, M. (Ed.), (1969), Semantic Information Processing, MIT Press (Cambridge, Mass.).

Minsky, M. (1990), "Logical verses Analogical, or Symbolic verses Conectionist, or Neat verses Scruffy", in Winston and Shelland (1990).

Minsky, M. (1992), Personal Communication by e-mail, September, 1994.

Minsky, M. and Papert, S. (1969), Perceptrons: An Introduction to Computational Geometry, MIT Press (Cambridge, Mass.). (3rd Edition published in 1988).

Mozer, M. and Smolensky, P. (1989), "Using Relevance to Reduce Network Size Automatically" in Connection Science, 1, pp. 3-16.

Newell, A. (1980), "Physical Symbol Systems", in Cognitive Science, 4, pp. 135-183.

Newell, A. (1982), “The Knowledge Level”, in Artificial Intelligence, 18/1, pp. 87-127.

Newell, A., Shaw, J. and Simon, H. (1957), "Empirical explorations of the logic theory machine: A case study in heuristics", in Feigenbaum and Feldman (1963).

Newell, A. and Simon, H. (1976), "Computer science as empirical inquiry: Symbols and search", in Communications of the ACM, 19/3, pp. 113-126.

Niklasson, L and van Gelder, T. (1994), “On Being Systematically Connectionist” in Mind and Language, 9/3, pp. 289-302.

Papert, S. (1988), "One AI or Many?", in Graubard (1988: pp. 1-14).

Parret, H. and Bouveresse, J. (Eds.), (1981), Meaning and Understanding, Walter de Gruyter (Berlin).

Pelletier, F. J and Berkeley, I. (1995), “Vagueness”, in Audi (1995: pp. 826-828).

Pinker, S. and Mehler, J. (Eds.), (1988) Connections and Symbols, MIT Press (Cambridge, Mass.).

Pinker, S. and Prince, A. (1988), “On language and connectionism: Analysis of a parallel distributed processing model of language aquisition” in Pinker and Mehler (1988: pp. 73-193).

Pitts, W. and McColloch, W. (1947), "How We Know Universals: The Perception of Auditory and Visual Forms," in Bulletin of Mathematical Biophysics, 9/127.

Pollack, J. (1990), "Recursive Distributed Representations", in Artificial Intelligence, 46, pp. 77-105.

Posner, M. (1989), Foundations of Cognitive Science, MIT Press (Cambridge, Mass.).

Pylyshyn, Z. (1984), Computation and Cognition, MIT Press (Cambridge, Mass.).

Pylyshyn, Z. (1991), “The Role of Cognitive Architecture in Theories of Cognition”, in VanLehn, (1991).

Quillian, R. (1969), "Semantic Memory", see Minsky (1969).

Quine, W. (1951), "Two Dogmas of Empiricism", reprinted in Martinich (1985: pp. 26 - 39).

Ramsey, W., Stich, S. and Rumelhart, D. (1991), Philosophy and Connectionist Theory, Lawrence Erlbaum Associates (Hillsdale, NJ).

Rips, L. J. (1983), "Cognitive Processes in Propositional Reasoning" in Psychological Review, 90/1, pp. 38-71.

Rips, L. J. (1994), The Psychology of Proof, MIT Press (Cambridge, Mass.).

Robinson, D. (1992), "Implications of Neural Networks for How We Think about Brain Function", in Behavioral and Brain Science, 15, pp. 644-655.

Rosenblatt, F. (1958), "The Perceptron, a Probabilistic Model for Information Storage and Organisation in the Brain", in Psychological Review, 62/386.

Rosenblatt, F. (1962), The Principles of Neurodynamics, Spartan (New York).

Ross, A. and Belnap, N. (1975), Entailment: The Logic of Relevance and Necessity, (2 Vols.), Princeton U. P. (Princeton, NJ).

Rumelhart, D. (1989), "The Architecture of Mind: A Connectionist Approach", in Posner (1989: pp. 133-159).

Rumelhart, D. and McClelland, J. (1987), "PDP Models and General Issues in Cognitive Science", pp. 110-146 of Rumelhart, McClelland et al. (1987).

Rumelhart, D. and Zipser, D. (1987), "Feature Discovery in Competitive Learning", in Rumelhart, McClelland, et al. (1987: Vol. 1, pp. 151-193).

Rumelhart, D., Hinton, G. and McClelland, J. (1987), "A General Framework for Parallel Distributed Processing", in Rumelhart, McClelland et al. (1987: pp. 45-76).

Rumelhart, D., Hinton, G. and Williams, R. (1987), "Learning Internal Representations by Error Propogation", in Rumelhart, McClelland et al. (1987: pp. 318-362).

Rumelhart, D., McClelland, J. and The PDP Research Group (1987), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, (2 Vols.), MIT Press (Cambridge, Mass.).

Ryle, G. (1949) The Concept of Mind, Barnes and Nobel (New York).

Schiffer, S. (1981), "Truth and the theory of content" in Parret and Bouveresse (1981: pp. 204-222).

Schneider, W. (1987), "Connectionism: Is it a paradigm shift for psychology?" in Behaviour Research Methods, Instruments & Computers, 19/2, pp. 73-83.

Seidenberg, M. and McClelland, J., (1989) "A Distributed, Developmental Model of Word Recognition and Naming", in Psychological Review, 96, pp. 523-568.

Searle, J. (1979), "Metaphor", reprinted in Martinich (1985: pp. 416-437).

Searle, J. (1980), "Minds, Brains and Programs" in Behavioral and Brain Sciences, III/3.

Searle, J. (1992), The Rediscovery of The Mind, MIT Press (Cambridge, Mass.).

Selfridge, O. (1959), "Pandemonium: A paradigm for learning", in Symposium on the Mechanisation of Thought Processes, HMSO (London).

Shannon, C. and McCarthy, J. (Eds.), (1956), Automata Studies, Princeton U.P. (Princeton, NJ).

Shastri, L. (1991), "The Relevance of Connectionism to AI: A Representation and Reasoning Perspective", in Barnden and Pollack (1991: pp. 259-283).

Shastri, L. and Ajjanagadde, V. (1993) ""From Simple Associations to Systematic Reasoning", in Behavioral and Brain Sciences, Vol. 16, No. 3 (Sept. 1993).

Smith, B. C. (1996), On the Origin of Objects, MIT Press (Cambridge, Mass.).

Smolensky, P. (1988), "On the Proper Treatment of Connectionism", in Behavioral and Brain Sciences, 11, pp. 1-74.

Smolensky, P. (1990), "Tensor Product Variable Binding and the Representation of Symbolic Structures in Connectionist Systems", in Artificial Intelligence, 46, pp. 159-216.

Smolensky, P. (1991), "Connectionism, Constituency, and the Language of Thought" in Loewer and Rey (1991: pp. 201-227).

Smolensky, P. (1994), "Constituent Structure and Explanation in an Integrated Connectionist/Symbolic Cognitive Architecture" in MacDonald & MacDonald (1994).

Sterelny, K. (1990), The Representational Theory of Mind, Blackwell (Oxford).

Stich, S. (1983), From Folk Psychology to Cognitive Science: The Case Against Belief, MIT Press (Cambridge, Mass.).

Stillings, N., Feinstein, M. Garfield, J., Rissland, E., Rosenbaum, D., Weisler, S. and Baker-Ward, L. (1987), Cognitive Science: An Introduction, MIT Press (Cambridge, Mass.).

Tienson, J. (1991), "Introduction", pp. 1-29 of Horgan and Tienson (1991).

Turing, A. (1936), "On Computable Numbers, with an Application to the Entscheidungsproblem," in Proceedings of the London Mathematical Society, 2/42, pp. 230-265.

Turing, A. (1950), "Computing Machinery and Intelligence", in Mind, LIX/236, pp. 433-460.

Twining, W. and Miers, R. (1976), How to Do Things with Rules, Weidenfield and Nicolson (London).

van Gelder, T. (1990), "Compositionality: A Connectionist Variation on a Classical Theme" in Cognitive Science, 14, pp. 355-384.

van Gelder, T. and Niklasson, L. (1994), "Classicalism and Cognitive Architecture", Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society, Atlanta, Georgia, pp. 905-909.

van Gelder, T. and Port, R. (1993), "Beyond Symbolic: Prologomena to a *Karma-Sutra* of Compositionality" in Honavar and Uhr, (1993). All page references to this paper are to a manuscript version.

VanLehn, K. (Ed.), (1991), Architectures for Intelligence, Lawrence Erlbaum Associates (Hillsdale, NJ).

von Neumann, J. (1951), "The general and logical theory of automata", in Jeffress, (1951: pp. 1-41).

von Neumann, J. (1956), "Probabilistic logics and the synthesis of reliable organisms from unreliable components" in Shannon and McCarthy (1956: pp. 43-98).

von Neumann, J. (1966), "Rigorous Theories of Control and Information", in Theory of Self-Reproducing Automata, U of Illinois Press (Urbana).

Waismann, F. (1951), "Verifiability" in Flew (1951: pp. 117-144).

Wall, R. (1995), "Grammar", in Audi (1995: pp. 302-303).

Winlow, W. (1990a), "Prologue; The 'typical' neurone", in Winlow (1990: pp. 1-4).

Winlow, W. (Ed.), (1990), Neuronal Communications, Manchester U. P. (Manchester).

Winston, P. and Shelland, S. (Eds.), (1990), Artificial Intelligence at MIT: Expanding Frontiers, (Vol. 1), MIT Press (Cambridge, Mass.).

Wittgenstein, L. (1953), Philosophical Investigations, Basil Blackwell (Oxford).